Urval

Skriftserie utgiven av statistiska centralbyrån

Nummer 13

Lars Lyberg

Stockholm 1981

Control of the Coding Operation in Statistical Investigations -Some Contributions

SCB STATISTISKA CENTRALBYRÅN

Urval

Skriftserie utgiven av statistiska centralbyrån

Nummer 13

Lars Lyberg

Control of the Coding Operation in Statistical Investigations -Some Contributions

SCB STATISTISKA CENTRALBYRÅN

UNIV.
LEIDEN
BIBL.

ACKNOWLEDGEMENTS

Several years ago I had the privilege of working as an assistant to Professor Tore Dalenius at the University of Stockholm. Among other things, I participated in his research project ERRORS IN SURVEYS (financed by the Bank of Sweden Tercentenary Fund), and it was here that I first encountered the problem of control of coding. Eventually I had the opportunity to carry on this work at Statistics Sweden, and my knowledge of this subject is summed up in this thesis, a summingup that has taken many years.

I want to express my deepest gratitude and my warmest thanks to Professor Tore Dalenius. Without his supervision, enthusiasm and great interest in my work this thesis would never have been written. His influence on my attitude towards survey methodology has been tremendous; over the years I have learned a great deal from him and he has become a very dear friend of mine.

My gratitude is extended to my wife, Mrs Ingrid Lyberg, Statistics Sweden, for many valuable discussions, for checking most of the formulas, for programming the functions used in chapter 12, and for maintaining a research-oriented atmosphere at our home in Spånga City.

I also want to thank fil.dr Bengt Swensson for his very helpful comments on a preliminary version of this thesis; it was much improved by his constructive critisism.

I am indebted to fil.lic Christer Arvas, one of my chiefs at Statistics Sweden, for his enthusiastic encouragement during the work. I am also indebted to Statistics Sweden for financial help:this agency has fully sponsored all typing, printing and computing in connection with the writing of this thesis. Statistics Sweden has also sponsored two trips to the US Bureau of the Census where I had the

opportunity of discussing coding problems with many colleagues. Of course, the coding problems in different surveys at Statistics Sweden have provided the necessary basis for this thesis.

During the years many other colleagues have taken a great interest in my work, none mentioned and none forgotten. I am deeply indebted to all of them.

I am also much obliged to Ms Katrina Avery, Brown University, for changing my Swenglish to American English. She has done a great job and any remaining faults are certainly due to lastminute changes of my own.

Finally, I am deeply indebted to Miss Monica Larsson, Statistics Sweden, for her very skillful typing. Her great patience, when presented with one version after the other, overcomes everything.

Stockholm, March 1981.

Lars Lyberg

CONTENTS

INTRODUCTION

1 "Survey" and "total survey design"
2 The coding operation and the characteristics of the control problem
3 The subject and organization of the thesis

Part I CODING, ERRORS AND CONTROL - AN OVERVIEW

4 Coding
5 Coding errors
6 Control of coding operations

Part II MANUAL CODING

7 Introduction to part II
8 The importance of "good input"

A PREVENTIVE CONTROL

9 More on the design of the code
10 Selection and training of production coders and verifiers

B PRODUCTION CONTROL

11 Dependent verification
12 Independent verification

APPENDIX I: Probabilities and costs for independent verification schemes

APPENDIX II: Comparing schemes

13 The extent of verification

APPENDIX III: Formulas for computing average cost per lot

C SOME ADDITIONAL TOPICS

14 Errors in coding: detailed discussion
15 Prior information when designing schemes for manual coding operation
16 Decentralized coding

APPENDIX IV: Coding processes
IV.1 The case of Sweden - a review
IV.2 Coding in population censuses in Sweden

Part III AUTOMATED CODING

17 The challenge

A THE MAIN COMPONENTS

18 A bird's-eye view of automated coding
19 The dictionary
20 The automated coding procedure

B EXPERIMENTS AND APPLICATIONS

21 Non-Swedish illustrations - an overview
22 Swedish illustrations - industry
23 Swedish illustrations - occupation
24 Swedish illustrations - goods
25 Swedish illustrations - book loans
26 Swedish illustrations - medicine

EPILOGUE

REFERENCES

INTRODUCTION

1 "Survey" and "total survey design"

In order to place the subject of this thesis - coding, coding errors and their control in surveys - in the proper context, we will discuss briefly the notion of a "survey" in section 1.1, and "total survey design" in section 1.2.

1.1 The notion of a survey

In Dalenius (1974) seven characteristics of a survey are presented:

(1) A survey concerns a well-defined object or set of objects of some kind, to be referred to as a population.

(2) This population has one or several measurable properties ("attributes").

(3) One wants to describe the population by one or more "parameters" defined in terms of these properties. This calls for observing (a sample from) the population.

(4) In order to get observational access to the population, a frame is needed: that is, an operational representation of some kind (for example, a register or a map of the population). The access to the population may be by means of one or both of the following techniques:

a) The frame is divided into (a hierarchy of) sampling units. There is a "rule of association" which associates each object uniquely with a sampling unit.

b) The frame is "continuous". Observations are to be made at a sample of points (or other geometrical entities, like lines, or areas) located in the frame.

(5) A sample of (ultimate) sampling units or points, etc. is selected from the frame in accordance with a sampling design which specifies a probability mechanism for selecting the sample and also the number of sampling units (points) to select. As a special case, the design may call for selecting all sampling units.

(6) Observations are made on the sample in accordance with a measurement design: a measurement method and a prescription as to its use.

(7) Finally, on the basis of the observations thus collected, an estimation design is applied in order to compute the quantities used as estimates of the "parameters" when making inferences from the sample to the population.

The term "survey" will be used here to denote any statistical study which can be characterized by (1) - (7) above.

1.2 Total survey design

Early sample survey theory and methods dealt mainly with the problem of measuring and controlling specific sources of errors. An important example is the construction of sampling models, i.e. models which do not take into account the existence of measurement errors. This area of study was very successful; a sizeable number of techniques are available today for controlling the sampling error (that part of the total survey error arising from the fact that only a sample of the population is observed).

Since the 1940s there has been an increasing awareness of the problem of non-sampling errors in surveys. As pointed out in Dalenius (1962), two lines of development have emerged in this work:

(1) Development of theory and methods for coping with specific sources of non-sampling errors.

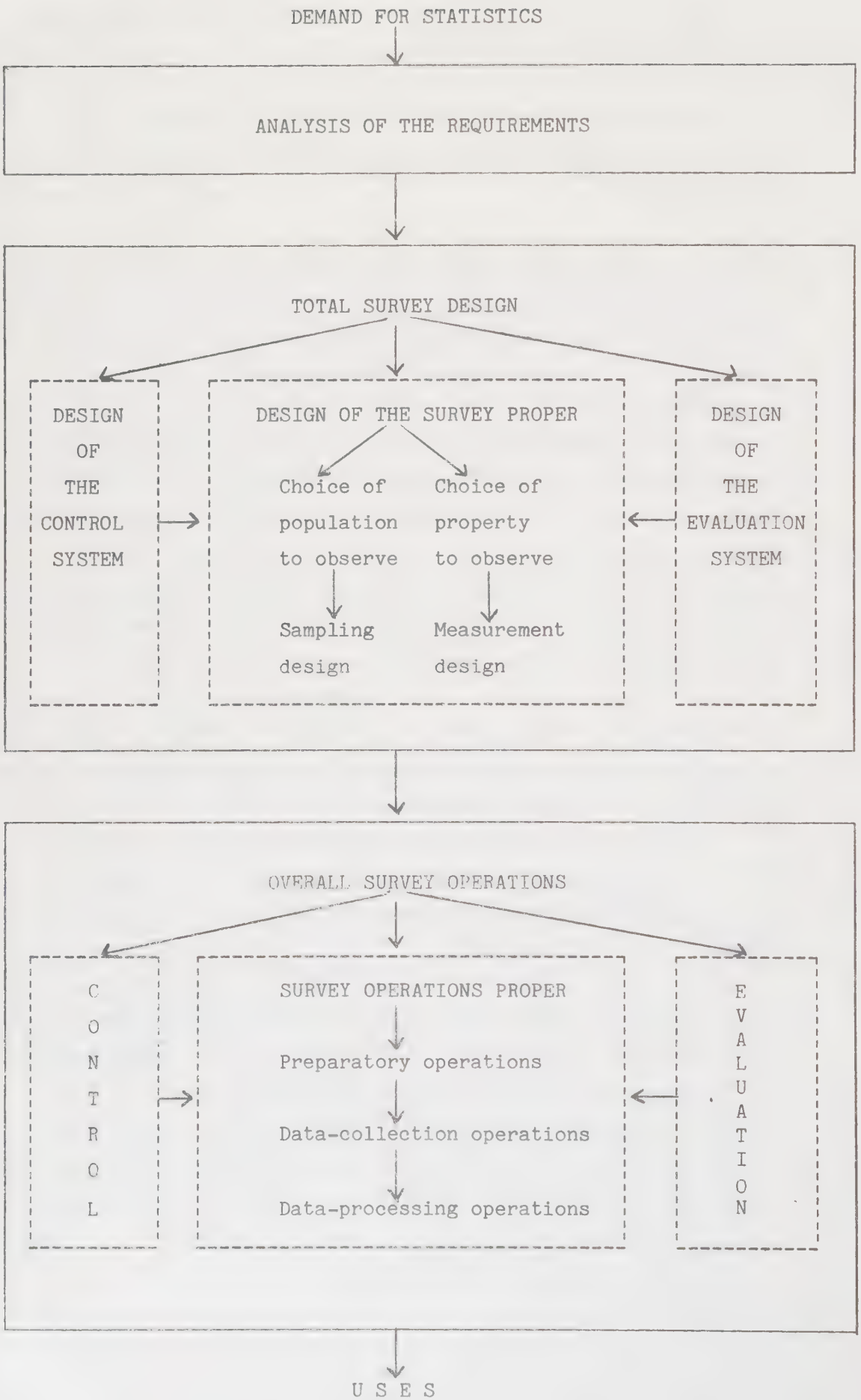
(2) Development of a comprehensive theory of non-sampling errors (or, in the terminology of the early 1960s, of mixed error models).

During the last decades great efforts have been made toward constructing what are called total survey models. These survey models are characterized, among other things, by the fact that they postulate the presence of measurement errors. One such model has been constructed at the US Bureau of the Census and is presented in Hansen et al.(1961, 1964). Another is discussed in Sukhatme and Seth (1952). A model developed at Statistics Canada is presented in Fellegi (1964) and still another constructed at the US National Center for Health Statistics is presented in Casady (1966). The techniques for randomized response, originating in Warner (1965), might also be mentioned as an example of efforts to construct total survey models.

Much of the later development in total survey design is given a systematic, comprehensive review in Dalenius (1974) which is based on the three fundamental perspectives:

i) the requirements
ii) the specification of the survey
iii) the survey operations

These perspectives are considered in combination with the control and evaluation of the survey part of the overall survey program. The resulting overview is given on the next page.


DEMAND FOR STATISTICS
ANALYSIS OF THE REQUIREMENTS
TOTAL SURVEY DESIGN
DESIGN OF THE CONTROL SYSTEM
DESIGN OF THE SURVEY PROPER
Choice of population to observe
Choice of property to observe
Sampling design
Measurement design
DESIGN OF THE EVALUATION SYSTEM
OVERALL SURVEY OPERATIONS
CONTROL
SURVEY OPERATIONS PROPER
Preparatory operations
Data-collection operations
Data-processing operations
EVALUATION
USES

This view of total survey design naturally leads to the following considerations:

a) Corresponding to the requirements, there is an ideal set Z of relevant statistics.

b) Due to practical limitations, however, the actual survey will yield a set y of statistics which differs from Z.

c) Thus a theory is necessary which will make it possible to measure, control and evaluate the total error y - Z.

d) The total error y - Z is measured by the mean square error $E(y - Z)^2$.

e) Suitable decompositions of the mean square error, together with cost considerations, will make possible efficient resource allocation for controlling the overall error of the survey.

The distinguishing character of a total survey design is, as pointed out in Dalenius (1974), that the problem of estimating Z is formulated as a measurement problem, in principle analogous to any measurement problem. Hence the term "survey measurement system" is often used in place of "total survey design".

2 The coding operation and the characteristics of the control problem

Examples of data-processing operations in a survey are editing, coding, key-punching and tabulation. Consider a collective of objects ("elements") of some kind and a set of mutually disjoint categories. Each element belongs to one and only one of these categories. Coding denotes the act of assigning the elements into these categories.

In practice the coding is based upon access to verbal information about the elements of the population or sample under study. This information is usually obtained on schedules in the data-collection operation and is entered either by the respondents themselves or by interviewers or enumerators. Unlike certain other kinds of information numerical data on household expenditures, for instance, verbal information cannot be processed immediately into statistical tables. It must first be coded into different categories where each category is labeled with, for instance, a number. These numbers are called code numbers and the key to these code numbers is called the code. (Naturally, numerical data also may be subject to coding; thus in a census of businesses the objects enumerated can be assigned to categories defined with respect to e.g. total turnover).

The term "coding" is admittedly ambiguous. Attempts have been made to replace it by the term "classification"; this term may be better than coding but it has certain disadvantages. Throughout this thesis the term "coding" is used since it is the one most frequently used in the literature. Some other terms in this area are ambiguous as well. For instance, what in this thesis is called "code number" is in the literature often referred to as "code", and what here is called "code" is often referred to as "code list", "coding standard", "lexicon", or "nomenclature". Still another ambiguity concerns <u>what</u> is to be coded. In the definition above it was postulated that a given element belonged to a certain category. In the literature

coding is often described as an operation in which the verbal descriptions or the responses are coded rather than the elements themselves. This common way of describing the situation is easily understood since in many surveys each element is coded with respect to more than one variable.

The coding operation has three components:

(1) Each element in, for instance, a population is to be coded with respect to a specific variable by means of verbal descriptions.

(2) There exists a code for this variable, i.e. a set of code numbers in which each code number denotes a specific category of the variable under study.

(3) There is a coding function relating (1) and (2), i.e. a set of coding instructions relating verbal descriptions with code numbers.

Coding is a major operation in such statistical studies as censuses of population, censuses of business and labor force surveys. Examples of variables are occupation, industry, education and status.

The problems with coding are of different kinds. As with most other survey operations, coding is susceptible to errors. The errors occur because the coding function is not always properly applied and because either the coding function itself or the code is improper. In fact, in some statistical studies coding is the most error-prone operation next to data collection. For some variables error frequencies at the 10 % level are not unusual. Another problem is that coding is difficult to control. Accurate coding requires a lot of judgment on the part of the coder, and it can be extremely hard to decide upon

the correct code number. Even experienced coders display a great deal of variation in their coding. Thus there are problems in finding efficient designs for controlling the coding operation. A third problem is that many coding operations are difficult to administer. Coding has a tendency to become time-consuming and costly: for instance, in the 1970 Swedish Census of Population carrying out the coding took more than 300 man-years. In many countries coders in large-scale operations must be hired on a temporary basis, and the consequences for maintaining good quality are obvious. There are even reasons to believe that in the future it might be difficult to obtain even temporary coders for this kind of relatively monotonous work. So there is certainly room for new ideas on the effectiveness of the coding operation.

3 The subject and organization of the thesis

In this thesis we will discuss coding and coding errors and their control within the conceptual framework presented in section 1 above.

The thesis consists of three parts.

Part I, CODING, ERRORS AND CONTROL - AN OVERVIEW, is intended to give a bird's-eye view of the characteristics of the coding operation and the problem of its control. It consists of three chapters. The first chapter deals with the coding operation itself: the notion of coding, a brief description of selected codes used at the Swedish National Central Bureau of Statistics[1], and the definition of some broad categories of coding operations, for instance manual vs automated coding. The second chapter deals with coding errors: after a discussion of the meaning of "coding error", review of some experiences is given. The third and final chapter presents the characteristics of the control problem. The need for control is stated and some basic control tools are described, viz. the US Bureau of the Census' survey model, statistical quality control schemes, some specific coding control schemes and automated coding.

Part II, MANUAL CODING, consists of ten chapters. Initially the importance of "good input" is stressed. The effects of good input is illustrated with a few examples. After that a couple of short chapters on preventive control are given, dealing with the design of the code and the selection and training of coders and verifiers. Then different production control measures are presented. Production control is associated with two basic problems: what kind of verification should be used, and how much. Two basic verification procedures, dependent and independent verification, are presented in detail.

---

[1] In what fallows, this agency will for brevity be referred to as "Statistics Sweden".

In the section called "some additional topics", some special problems associated with manual coding are attacked. First a more detailed illustration of errors in coding is given. Most of that section deals with evaluation of coding, which can have an impact on preventive measures taken on future coding operations. After that some ideas concerning the use of prior information when designing control operations are put forward, and the method of decentralized coding is described. Part II ends with an appendix on coding processes. So far different techniques have been discussed without special emphasis on the feasibility of carrying them out. In a practical situation the statistician has to choose between different means and often some of them have to be abandoned because of financial or administrative reasons. The practioner usually has to resort to a limited number of techniques, sometimes even modified, and this results in coding processes which are far from ideal. Here some "real" coding processes are described, with emphasis on Swedish population censuses.

Part III, AUTOMATED CODING, deals with automated coding. Automated coding could be used as a control technique as well as a means for obtaining more effective coding in general. Automation of the coding process is a real challenge. Manual coding is costly and time-consuming, and computers have found applications in other related fields, for instance editing.

Part III consists of ten chapters. First the challenge itself is presented. After that an overview of the main components of an automated coding scheme is given: constructing a dictionary, entering descriptions into the computer, matching and coding, and evaluation. These topics are then treated in detail, the dictionary component in a special chapter and the rest of the components in a chapter on the automated coding scheme. The rest of part III deals with applications of automated coding,

both foreign and Swedish. Swedish applications mainly concern the variables occupation and goods; with respect to goods, automated coding has been applied on a large scale in the 1978 Swedish Household Expenditure Survey.

The EPILOGUE, finally, discusses the increasing use of administrative records in statistics production. It also discusses possible methodological developments in the computer field and the future of automated coding.

Much of the discussion in Parts II and III makes use of empirical studies of various kinds. Some studies exploit data which have been obtained as by-products of ongoing surveys. Other studies are in the nature of experiments imbedded in surveys, or are carried out separately (although in a production coding environment). Still other studies are based on "laboratory experiments" (conducted in an environment differing from production coding). In each case it should be clear to the reader what kind of experiment we are referring to.

The statistical investigations used for this thesis are characterized by a lack of variance estimates (and hence calculation of confidence intervals). One reason for this state of affairs is that many of the individual investigations are based on rather small samples, which produces large variances for the individual estimates. These investigations have, however, been viewed simultaneously and have been analyzed in the spirit of Tukey's phrase "borrow strength ...".

In this thesis it happens that the same techniques are discussed in different chapters and sections. This has been unavoidable due to the fact that various techniques must be discussed in presenting various coding processes. Moreover the thesis is meant to be self-contained to make it easier for the reader to remember all details. As a consequence some repetition has been necessary.

Finally, the use of "we" in this thesis should be explained. The thesis has been developed in the course of the author's work at Statistics Sweden. To some extent, this work has been carried out as "team work". When a significant contribution has been made by someone other than the author, this is typically indicated by giving a reference to documented work by that other person, as "Bäcklund (1978)".

Part I CODING, ERRORS AND CONTROL - AN OVERVIEW

4 Coding

In this chapter, we will discuss the meaning of "coding" as used in this thesis (the term has a different meaning in information theory and related fields).

4.1 The notion of coding

First let us illustrate coding with a simple example.

Household member	Sex
Ingrid Lyberg	Female
Lars Lyberg	Male
Luis Lyberg	Male
Calle Lyberg	Male

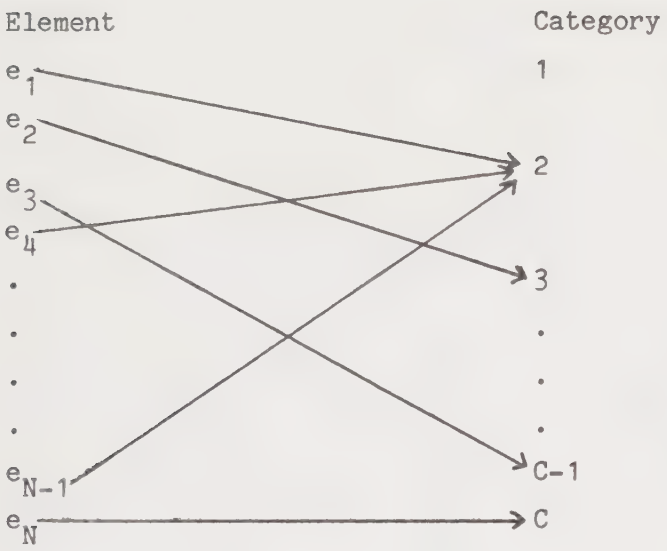
These household members are to be coded as to sex; i.e., the coding operation amounts to assigning the four members into two categories denoted by code numbers, say, "1" for female and "2" for male. Thus, the coding yields:

Household member	Sex	Code number
Ingrid Lyberg	Female	1
Lars Lyberg	Male	2
Luis Lyberg	Male	2
Calle Lyberg	Male	2

A more general description of coding is the following. Let us assume that we are going to study a collective of N elements of some kind denoted by e_1, e_2,..., e_N. Each of these elements belongs to one and only one of, say, C mutually disjoint categories. Coding is the act of assigning the elements into these categories.

This general view of coding can be illustrated by the following figure.

Figure 1



In this figure, element e_1 has been assigned to category 2, element e_2 to category 3, etc. In practice the categories are often labeled with other numbers depending on the structure of the code.

Usually each category is described ("defined") in terms of short key-words but sometimes in more detail. Defining a set of categories is motivated by the fact that producers and users of statistical data want to compress the information collected for the different elements of the population. The reason is quite straightforward. Usually verbal information about the elements is obtained on schedules. The input space is, in principle, made of a more or less free-format set of verbal descriptions which cannot be catalogued in advance because the number of possible descriptions or statements is extremely high; thus the input space is for all practical purposes a space with an infinite number of points (possible descriptions).

The code, the nomenclature, the classification standard or simply the dictionary are different terms for the same thing: the set of code numbers with associated definitions of each category. The dictionary is used to code elements such as individuals, companies and farms as to specific variables such as occupation, number of employees, and crop acreage respectively.

Hyvärinen (1971) presents an instructive formalization of the coding process[1]. Thus, let s denote a description (with respect to some variable (characteristic) of interest, such as sex in the example above) of an element. This description is but one possible in a large space of descriptions S, i.e. $s \in S$. The coding procedure is a transformation T of s, say T(s), which yields a point c in a space C of code numbers:

(1) $T(s) = c \; ; \; s \in S \; ; \; c \in C$

In practice the size of C is much smaller than the size of S, so (1) is a many-to-one transformation.

A good transformation should map all descriptions s that have the same meaning, and only those descriptions, into one point in C (for instance, the same occupation into the same code number).

4.2 Selected codes used by Statistics Sweden

The codes used by Statistics Sweden are descriptions of the possible categories for the variable under study. Each category is identified by a code number. Thus ideally the coder looks at the element description of the variable, consults the code and assigns the proper code number. The complexity of codes varies. At one extreme we have the code for sex which, in principle, consists of only two categories. At the other extreme we have, for instance, the complex code for causes of death,

---

[1] An alternative presentation uses the notion of "fuzzy sets", as discussed in e.g. Zadeh (1965).

which contains several thousand categories. The more complex codes are often built hierarchically, like Chinese boxes: for instance, the first digit gives a major group, the second digit gives a group within this major group, and so forth. As a consequence variables with fewer categories are handled by means of fewer digits. One often speaks about one-digit, two-digit or multi-digit variables. The code is usually constructed by producers and users of statistics together.

In this section some of the codes used by Statistics Sweden will be briefly presented. Different codes are described in various publications, and there are also comprehensive descriptions covering large parts of the entire set of codes used in official statistics production. A description of the latter kind is found in Statistiska centralbyrån (1973). Note that in this publication the word "standard" is used instead of "code" but it should be pointed out that there are still other terms as well. A few examples are "nomenclature", "dictionary", "classification standard" and "coding scheme".

i) SNI

SNI stands for "Swedish Standard for Classification of Economic Activities". This standard is used when coding the variable industry and it has six hierarchical levels, four of which are nearly identical with its international counterpart ISIC. SNI comprises about 370 different industry categories on the four-digit code number level and is used for instance in censuses of population, the labor force survey, surveys of enterprises carried out by Statistics Sweden and in certain administrative records. Statistics Sweden has also recommended that other Swedish agencies use it when dealing with the industry variable. For instance, the local taxation authorities use the SNI on the two-digit level.

The first digit in the code gives the main industry, the second gives an area within this main industry, etc. Let us illustrate the principle with the following example.

First digit	3	Manufacture
Second digit	36	Manufacture, goods from mineral substances except metals
Third digit	369	Manufacture, bricks and cement
Fourth digit	3699	Manufacture, other mineral goods
Fifth digit	36992	Manufacture, concrete and concrete goods
Sixth digit	369922	Manufacture, light concrete goods

The code is further described in Statistiska centralbyrån (1969 b).

ii) SUN

SUN is the "Swedish Standard for Classification of Education". It has five different levels consisting of 2400 individual branches of education (five-digit code numbers) and 300 categories for type of education (three-digit code numbers). It has been used in population censuses and is currently used, for instance, in surveys of pupils and in certain labor market surveys. The code is described in SCB (1969 a).

iii) ICD

ICD is the "Standard for International Classification of Diseases". It is used in disease and death statistics as well as in classification of diagnoses in patient files, etc. It consists of hierarchical three-digit code numbers with subgroups on decimal levels. The code is described in World Health Organisation (1967).

iv) NYK

NYK is the "Northern Standard for Classification of Occupations" built upon the "International Standard for Classification of Occupations" (ISCO). The purpose of this code is to simplify employment exchange within the northern labor market. Furthermore, it should make possible to gather occupational statistics for the different countries. NYK is designed hierarchically with three different levels (three-digit code numbers). The categories on the two highest levels are almost identical with the corresponding categories in ISCO. The code also contains a detailed level used for special occupations (five-digit code numbers). It is used, for instance, in population censuses, the labor force survey and in pupil surveys. The code is described in Statistiska centralbyrån (1965) and is under revision.

4.3 Broad categories of coding operations

Coding can be carried out manually outside an agency, manually within an agency, or automatically by a computer.

Manual coding outside an agency is more common than one might think. In the continuous Swedish Labor Force Survey the coding of occupation is done by interviewers. In the 1975 Swedish Census of Population local authorities coded some of the census variables in order to make it possible to produce some employment statistics without the usual time lag. A third example is when the respondents themselves code different variables. At one extreme we have the case when respondents receive a copy of the code and are asked to use it for coding purposes. At the other extreme we have the case called "self-coding" where the respondent is presented with a number of fixed alternatives and asked to choose one of them.

Coding outside an agency shifts the burden of coding. Generally the procedure generates low coding costs for the agency but the control of the agency over the coding is reduced as well.

The literature on evaluation of this kind of decentralized coding is not very extensive.

Manual coding inside an agency (centralized coding) is very common. At Statistics Sweden such coding is used in over 100 surveys each year. However, in many of these cases, the coding operation is small and presents no serious problems. In other and more interesting cases, we have large-scale coding on a continuous basis. Examples are the censuses, in which Statistics Sweden during the last decades has hired around 300 coders on a temporary basis for each census. The coding operation is complex and many variables must be coded for each element. Such operations put tremendous pressures on the central staff and the organization. Examples of such operations are continuous surveys where variables such as occupation, industry, education and employment status are coded. Such surveys are the Labor Force Survey (both centralized and decentralized coding), different Pupil Surveys, Income Distribution Surveys, etc. The coding in these surveys is done by a regular coding staff.

Large-scale coding is costly and time-consuming and computer coding could be an attractive alternative. Automated coding is used as a complement to manual coding: manual coders take care of cases rejected by the computer. The methods for automated coding have been used experimentally for some years. Quite recently it was applied for the first time in production, in the 1978 Household Expenditure Survey where the variable "goods" was coded by computer. An immediate successor was the Swedish Author's Fund where authors and book titles were coded.

These different types of coding operations generate specific problems. In a decentralized coding situation there are problems involved in supervising and controlling the operation. In a centralized situation one often faces complex coding and has to make compromises between quality and cost. In computerized coding one must administer parallel manual systems and be very alert with respect to the performance of the computer programs.

Thus the main dimensions of the coding operations are

a) manual vs. computerized;
b) centralized vs. decentralized.

In practice, combinations of these dimensions may be used.

5 Coding errors

5.1 The meaning of "coding error"

In this thesis it is assumed that a true code number exists for each element with respect to the variable under study. A coding error occurs if an element is assigned to a code number other than the true one. This seemingly simple definition needs some further elaboration. Three different aspects will be considered.

First, it is often difficult to decide upon a true code number. The basic assumption is that every element belongs to one and only one category. In practice there are difficult situations where a specific description is such that it can be assigned different code numbers depending on interpretation.

Second, even if the description is detailed problems might arise in assigning true code numbers. Who are the experts to decide them? Studies show that the variation between "experts" can be considerable and as a consequence true code numbers often have to be defined operationally. For instance, three or more experts code the same element. By means of majority rule a true code number is assigned; i.e., the code number assigned by a majority of the experts is considered the true one.

Third, consider the following example. A dentist fills out a mail questionnaire. One of the questions is "What is your occupation?". For some reason or another the dentist answers "brain surgeon". Thus the information available to the coder is "brain surgeon"; if the code number for "brain surgeon" is, say, 411 and the coder assigns 411 then the coding is correct. If any other code number is assigned, including the one for dentist, a coding error has occurred. Thus we say that the coding operation starts with the available element description,

whether or not it is proper, and ends with the assignment of a code number. This limitation is practical from the standpoint of control. The obvious response error in the example must be dealt with by other means.

When the code has two or more digits, the notion of a coding error must take this fact into account. Assume for example that six coders code a specific element with respect to the variable "industry" using a 6-digit code. The outcome may be as follows:

True code number		3 6 9 9 2 2
Coder	1	3 6 9 9 2 1
	2	3 6 9 9 1 1
	3	3 6 9 1 1 0
	4	3 6 2 0 9 0
	5	3 5 5 1 1 0
	6	2 1 0 0 0 0

According to the definition of "coding error" all six code numbers assigned are wrong since none of them coincides with the true one. However, the errors are of different kinds. The first code number, 3 6 9 9 2 1, differs from the true one with respect to the last digit. The second coder has been able to code correctly down to the fourth level. The third, fourth and fifth coders have been able to code correctly down to the third, second and first level respectively. The sixth coder has not been able to assign the first digit correctly, and as a result the element has been coded to the main group "mining" instead of "manufacture". The point is that as soon as an error occurs on a specific level all subsequent levels are erroneously coded as well. Furthermore, as a consequence of the way an x-digit code is constructed, an error in the first position is more serious than an error in the second position which, in turn, is more serious than an error in the third, etc. For instance, an error in the sixth position only does not affect the quality of a presentation of results on the fifth level, but an error in the first position affects the presentation of results on any level.

The error rate is the number of incorrectly coded elements divided by the total number of elements coded. The error rate can be calculated for a specific material, for a specific time period, for individual coders and for different levels of the code.

Error rates are gross errors. Normally the results of the coding operation are displayed in statistical tables. The coding errors which remain in the table are net errors and can sometimes be much less than the gross errors, since the errors tend to some extent to cancel out. To see this, let us look at the following extreme example. Three men are living in a small village. One is a plumber, another is a sailor, and the third is an author. A complete census is carried out and the coding process unfortunately gives the following results: the plumber is coded as sailor, the sailor is coded as author, and the author is coded as plumber. Thus the error rate is 100 percent. If the census results are tabulated like this:

Number of plumbers	1
Number of sailors	1
Number of authors	1

the net effect of the coding errors is zero. However, this is by no means the general situation. Systematic coding errors can seriously distort statistical tables.

5.2 Some experiences

There is ample evidence that the coding operation is rather susceptible to errors. Error rates between 10 and 20 percent when coding variables such as occupation and industry are not uncommon.

Coding variability between and within coders can be substantial as well. The size of the between-coder variability depends heavily on the number of coders involved but there are examples of within-coder variabilities - in situations where a specific coder is presented with the same set of elements twice (with an intermission between trials in order to avoid memory effects) - exceeding 10 percent.

One situation in which coding errors always are serious is the following one. Suppose that we want to investigate people with special occupations for a health study. We have at our disposal a census file including all individuals in the area. But we are interested only in miners, stone-cutters and house-painters in our study of, say, pulmonary diseases. By means of the code numbers associated with these three occupations it should be possible to screen the appropriate population. Now the drawback associated with incorrect coding becomes obvious. When investigating the screened subpopulation we notice that some of the people under study do not in fact belong to the three occupation categories. The removal of those elements is a financial and administrative problem. Much worse is the fact that many miners, stone-cutters, and house-painters are hiding in the rest of the file under false code numbers.

These general experiences have occurred on a global scale. In this section we will give some examples from Statistics Sweden and a brief note on experiences of the US Bureau of the Census.

5.2.1 Some Swedish experiences

5.2.1.1 An example from a Labor Force Survey

An early study in Sweden concerning coding errors (Olofsson (1965)) treated the coding variability for the variables occupation and industry. It was found that coding errors seriously affected the estimation of parameters for gross changes, i.e., the flow between different categories. Some main results were that

i) only 40 percent of the changes in major (one-digit) occupation categories were real; the rest were coding errors. The corresponding figure for industry was 46 percent.

ii) only 30 percent of the changes in two-digit occupation categories were real. The corresponding figure for industry was 34 percent.

Obviously, publishing such estimates would indeed create an exaggerated picture of the mobility of the labor market. In fact, for some categories these coding errors lead to an overestimation of 100-200 percent.

5.2.1.2 The 1965 Swedish Census of Population

In 1967 an evaluation study of coding errors in the 1965 Swedish Census of Population was conducted (see Lyberg (n.d.) and Dalenius and Lyberg (n.d.)). From a population of census material comprising about 70 percent of the 1965 population a two-stage sample of verified census schedules was selected. The population was partitioned into four strata and four subsamples were obtained. The evaluation study was confined to the following four variables:

(1) Relationship to head of household
(2) Typ of employment
(3) Status
(4) Industry

The codes used for variables 1-3 were one-digit-codes; the code used for "industry" was a three-digit-code.

Since we were dealing with four variables and four subsamples we obtained 16 different estimates of error rates. These are given in Table 5.1 below. The four strata reflected the organization of the census operations. Thus subsample 1 consisted

of large city schedules subjected to total verification. Subsample 2 consisted of large city schedules subjected to sample verification. Subsample 3 consisted of totally verified schedules from other areas and subsample 4 consisted of sample verified schedules from these other areas.

Table 5.1. Estimates of error rates (%) in production coding in the 1965 Census of Population.

Subsample	Variable			
	1	2	3	4
1	1.6	2.7	1.0	14.5
2	1.4	1.6	.6	8.2
3	1.5	3.0	1.3	14.5
4	.7	1.3	1.2	8.7

Most of the total verification was done for still inexperienced production coders; this explains the differences in error rates between total and sampling verification.

In Lyberg (n.d.) studies concerning within-coder replication in the evaluation of the 1965 Swedish census are presented. Each of the three coders X, Y and Z in the experiment made one original coding (trial 1). After three weeks the material was coded once again by the same coders (trial 2). These independent trials gave the estimates shown in Table 5.2 of the within-coder variability $P = m/n$, where n is the total number of coded elements for the specific variable and m is the number of elements differently coded when comparing the two trials.

Table 5.2. Estimates of within-coder variability, $\hat{P}$ (%) in the 1965 Swedish Census of Population

Variable	$\hat{P}$ for coder			Subsample
	X	Y	Z	
Relation-	.5	.8	0	1
ship to	.6	.9	.6	2
head of	1.0	.5	.2	3
household	.5	.2	.5	4
Employment	1.8	2.1	1.6	1
	1.2	.9	.3	2
	1.0	2.2	.5	3
	.9	1.1	1.1	4
Occupa-	1.0	1.3	.5	1
tional	1.2	.6	0	2
status	.5	1.5	.5	3
	.7	.5	.7	4
Industry	7.4	5.7	4.5	1
	6.5	5.2	3.2	2
	6.7	1.8	1.8	3
	5.8	5.3	2.9	4

Except for industry, the variables are one-digit. The variability is rather modest for the three simple variables, with a maximum of 2.2 percent for coder Y and the employment variable. Industry variability ranges from 1.8 to 7.4 per cent; this is to be expected compared to the simple variables, but illuminates the difficulty of coding the more complex variables.

5.2.1.3 The 1970 Swedish Census of Population

In the 1970 Swedish Census of Population the number of variables to be coded increased over that in 1965. For evaluation purposes a sample was drawn from the population of census schedules. A pool of expert coders was used to generate a set of "true"

evaluation code numbers for each schedule in this sample. These code numbers were compared with the production code numbers after verification, and this led to estimates of error rates for the different variables on economic activity. These variables were

(1) Relationship to head of household
(2) Type of activity
(3) Occupation
(4) Status
(5) Industry
(6) Place of work
(7) Type of conveyance to place of work
(8) Number of hours at work

Estimates of error rates for these variables are given in Table 5.3.

Table 5.3. Estimated error rates in coding economic activity in the 1970 Swedish Census of Population

Variable	Code	Percent error rate (total population)
(1)	1-digit	4.3
(2)	1-digit	4.7
(3)	3-digit	13.5
(4)	1-digit	3.7
(5)	4-digit	9.9
(6)	1-digit	8.9
(7)	1-digit	11.5
(8)	1-digit	4.4

The error rates for the variables (1), (6) and (7) are probably overestimated, since the code numbers were processed by an optical character recognition machine and we have reason to believe that technical errors in this phase had a minor effect on the error rates for those variables.

The table shows that the multi-digit variables are difficult to code. But also the one-digit variables, a priori considered easily coded, are erroneously coded relatively often. One reason could be that the coding situation is too complex for one coder; i.e. each coder has more variables to manage than he can handle.

In the evaluation of the 1970 census coding the experiments for investigating the within-coder variability were repeated. This time five expert coders were used. The results are given in Table 5.4 below.

Table 5.4. Estimates of within-coder variability (%) in the 1970 Swedish Census of Population.

Variable	Expert coders				
	A	B	C	D	E
Relationship to head of household	.7	1.2	2.4	1.1	.8
Type of activity	1.2	2.1	3.0	1.5	1.8
Occupation	8.0	10.6	10.9	9.2	7.1
Occupational status	2.4	.9	1.8	1.1	1.9
Industry	3.7	8.8	11.6	6.9	5.4
Employment	1.6	3.2	3.9	2.7	2.1
Type of conveyance to place of work	.8	2.7	6.0	1.4	2.9
Place of work	1.3	1.5	2.1	1.8	2.5

The estimates are based on sample sizes ranging from 300 to 1000; this range reflects the fact that the economic variables are coded only for economically active persons. There were also some differences in expert coder workload.

As Table 5.4 shows the within-coder variability is considerable. This is disturbing when we remember that these coders were used as producers of "true" code numbers to evaluate the coding operation.

The between-coder variability can be obtained as a byproduct of a verification process. If more than two coders are involved P as defined in 5.2.1.2 could be replaced by

$$P' = 1 - \frac{m'}{n}$$

where m' is the number of elements for which all coders involved agreed upon a specific code number. The use of P' has a deceptive effect, however, since it depends on the number of coders involved.

In some experiments during the evaluation of the coding in the 1965 Swedish census P' took values from 13.0 to 14.1 per cent for three independent coders coding industry.

In some experiments prior to the 1970 census similar results were obtained. The results are presented in Lyberg (1970).

During the evaluation of the 1970 census coding we used five coders in a between-coder-variability study. These five coders independently coded 483 individuals. The variables were those in Table 5.4 above plus "number of hours at work" and three geographic variables. When using five coders the following combinations might occur:

5-0, 4-1, 3-2, 3-1-1,
2-2-1, 2-1-1-1,
1-1-1-1-1,

where 2-2-1, for instance, means that two coders have agreed upon a specific code number, two others have agreed upon another specific code number and the fifth has assigned a third code number.

The results are shown in Table 5.5 below.

Table 5.5. Absolute between-coder variability distributed over coding combinations and variables in a 1970 Swedish census study.

Coding combination / Variable	5-0	4-1	3-2	3-1-1	2-2-1	2-1-1-1	1-1-1-1-1	Number of elements
Relationship to head of household (1)	465	15	3	0	0	0	0	483
Type of activity (2)	436	32	9	6	0	0	0	483
Status (3)	179	18	7	0	0	0	0	204
Occupation (4)	146	35	12	5	3	3	0	204
Industry (5)	157	31	10	3	1	2	0	204
Number of hours at work (6)	180	18	5	1	0	0	0	204
Type of conveyance to place of work (7)	176	17	8	1	2	0	0	204
Place of work (8)	168	27	8	0	1	0	0	204
Geographic code I (9)	194	7	3	0	0	0	0	204
Geographic code II (10)	197	2	5	0	0	0	0	204
Geographic code III (11)	197	2	5	0	0	0	0	204

The between-coder variability is apparent. Let us define our P' as

$$P' = (1 - \frac{N_{5-0}}{N}) \ 100$$

where N_{5-0} is the number of cases for which all five coders agreed upon a specific code number and N is the total number of elements. Note that N is either 483 or 204, since only some individuals are economically active. The different values of P' are presented in Table 5.6 below.

Table 5.6. Estimated between-coder variability (%) per variable in a 1970 Swedish census study.

Variable	$\hat{P}'$
(1)	3.7
(2)	9.7
(3)	12.3
(4)	28.4
(5)	23.0
(6)	11.8
(7)	13.7
(8)	17.6
(9)	4.9
(10)	3.4
(11)	3.4

These P'-values are not very informative, since a 100 % variability can be expected if the number of coders is permitted to increase without restriction. Furthermore, these simple P and P' indexes can only be used for each variable separately. A P-value for variable X can seldom be compared to a P-value for variable Y since the index is biased: P obviously tends to increase with the number of possible categories. Thus we cannot compare P-values to say that the coding of "relationship to head of household" is better than that of "industry".

5.2.1.4 The 1975 Swedish Census of Population

The number of variables was smaller in the 1975 Census of Population than in the 1970 Census. Evaluation studies show that the error rates also were smaller in this census than in the 1970 Census. The following variables were studied:

(1) Relationship to head of household
(2) Type of activity
(3) Occupation
(4) Status
(5) Industry
(6) Type of employment
(7) Type of conveyance to place of work

All of these are one-digit variables except for (3) and (5). In Table 5.7 estimated error rates are given for these variables.

Table 5.7. Estimated error rates in coding economic activity in the 1975 Swedish Census of Population.

Variable	Code	Percent error rate (total population)
(1)	1-digit	.6
(2)	1-digit	.6
(3)	3-digit	7.8
(4)	1-digit	.5
(5)	4-digit	3.5
(6)	1-digit	1.0
(7)	1-digit	.5

The results given in this table differ strikingly from those obtained in the 1970 evaluation study. The error rates have dropped for every variable and the fact that the one-digit variables now really seem to be easily coded is most encouraging. The occupation error rate of almost 8 percent is still very serious, but compared to the 13.5 percent rate in 1970 it is a good result. Even better is the estimate for industry.

The possible reasons for these improvements will be discussed in subsequent chapters.

5.2.1.5 Some other studies of error rates at Statistics Sweden

Most of the coding studies at Statistics Sweden have been carried out within the censuses. This is rather natural since the coding is a very extensive operation in a census. During recent years interest in coding errors has grown and as a result some evaluation studies have been carried out in other surveys as well. Here some estimates of coding errors from such studies are given.

In Olofsson (1976) an industry error rate of 5.7 % is noted in the 1974 Labor Force Survey. Occupation in the same survey had an error rate of 6.2 %. In Harvig (1973 b) an 11 % error rate in occupation coding is estimated for coding data for university graduates. In Harvig (1973 a) a 3.2 % error rate is estimated when coding underlying causes of death. In Lyberg et al. (1973) an 8 % error rate is estimated when coding teacher's education. In this case the 95 % confidence interval was 5.9 % - 10.3 %.

An unusual but very nice result was noted in Harvig (1975 a) when the error in geographic coding in a survey of nurses was estimated: no errors were found! In Harvig (1975 b) a 2.6 % error rate was noted in coding occupation for adult students. This good result is, however, really a matter of interpretation. The denominator of the frequency is the number of all adults in the survey. Many of these do not have an occupation and thus the coding is simple. An estimate based on only those with an occupation would have given a more realistic picture. Finally, in Harvig (1975 c) an evaluation study of the coding in the Survey of Living Conditions is described. The error in industry coding on the two-digit level is an estimated 1.3 %, for union membership 2.4 %, and for occupation on a very restricted basis (only 15 available code numbers) 2.7 %.

5.2.2 A note on the experience of the US Bureau of the Census

At the US Bureau of the Census, interest in coding problems has existed for many years. As early as the 1950 Censuses of Population and Housing an extensive quality control program was part of the coding operation.

In US Bureau of the Census (1974) the extensive coding evaluation program for the 1970 Census of Population is described. The estimate of the error rate for industry coding in that census was 9.1 % and for occupation 13.3 %. These results coincide almost exactly with those obtained in the Swedish 1970 Census. In a recent pretest for the 1980 Census, experienced Current Population Survey coders had an estimated error rate of 6.9 % when coding industry and an estimated error rate of 8.1 % when coding occupation. The latter experiment is described in US Bureau of the Census (1977).

6 Control of coding operations

6.1 The need for control

We have noted that error frequencies can be substantial; i.e. the gross errors can be large. Such errors may or may not have a significant impact on survey results (for instance estimates in statistical tables). The general experience seems to be that coding errors do not affect tables of overall statistics very much, since gross errors have a tendency to cancel out and become rather small net errors.

Statistical tables on overall statistics are seldom, however, the single and final output from surveys. Statistics in breakdowns may be seriously in error even if overall statistics are not. Besides, most surveys are multi-purpose and coded materials are often saved for future known or unknown analyses. It is common that a coded material with large gross errors is presented as a frequency distribution, say N_1, N_2,..., N_k for K categories, where the net effects of coding errors are small. After some time it is decided that a new survey or a special analysis should be carried out for individuals belonging to one or a few specific categories. Now, the gross errors may become serious as discussed above.

Other difficult situations occur when the material is used in cross-classification or in prediction (as was the case in the labor market mobility example in chapter 5).

These problems make it imperative in most statistical series that coding control be part of the overall program for producing the statistics. However, knowledge of the error rate is not enough if we want to be far-sighted. We need to know about the error structure, the reliability of the coding process, different types of errors, the seriousness of different errors and the effects of errors, in order to take suitable corrective measures with respect to the code or the coder.

It should be pointed out, though, that the control called for must be designed so as to take it into account that there are also other sources of errors. What is called for is a rational balance of various control efforts, as discussed in Dalenius (1974).

Some tools for the necessary coding control will be reviewed in sections 6.2-6.4.

6.2 The US Bureau of the Census' survey model

As pointed out in Hansen et al. (1964), the development of theory and methods for the measurement of mean square error (MSE) of survey results serves at least three purposes. One is to provide the survey designer with input for improving his work. A second is to control the survey process in order to provide assurance that it operates in accordance with given specifications. A third is to evaluate the survey results so as to provide the users with information to guide them in making decisions based on the survey results.

It is beyond the scope of this chapter to elaborate on the survey model presented in Hansen et al. (1964); we will be satisfied with a brief discussion of the decomposition of the MSE. The population mean $\bar{X}$ is estimated by $\bar{y}$, say by means of a simple random sample of elements. Then the total error $\bar{y} - \bar{X}$ is measured by $MSE(\bar{y})$, which can be written as:

$$MSE(\bar{y}) = \frac{\sigma_S^2}{n} + \frac{\sigma_R^2}{n}[1 + (n-1)\rho_R] + \frac{2(n-1)}{n}\sigma_{RS} + B^2$$

The first term is the sampling variance, the second is the total response variance, the third is the covariance of the response and sampling deviations and the fourth is the squared bias. It is important to remember that the sampling variance measures variations induced by the sampling process, while

the response variance measures variations assumed to characterize the measurement operation. An important feature of the model is its broad applicability: it may be applied to any sequence of survey operations, i.e. either the full sequence or a subset of operations (for instance, interviewing and coding). Applied to the full sequence, the response variance reflects contributions from all operations such as interviewing, coding, editing and so forth. Applied to coding alone, for instance, the response variance reflects only coding and the response variance becomes a coding variance. Consequently, coding gives a contribution to MSE of the form

$$\frac{\sigma_C^2}{n}\left[1 + (n-1)\,\rho_C\right] + B_C^2$$

The variance consists of a simple coding variance and a correlated coding variance.

In US Bureau of the Census (1972 b), Jabine and Tepping (1973) and Bailar and Dalenius (1969) it is described how the variance components of MSE($\bar{y}$) may be estimated. Essentially the estimation is carried out by means of replications and interpenetrations. Bailar and Dalenius develop a set of basic study schemes in order to obtain observations useful for the estimation process. In Hartley and Rao (1978) a general procedure is provided for estimating the total variance, including coding variance, directly from the current survey data. Naturally, this procedure requires special survey designs involving some kind of imbedded experiments. The Hartley-Rao procedure is improved in Biemer (n.d.) and the latter also incorporates a coder allocation scheme.

Thus the US Bureau of the Census' survey model has two roles with respect to coding control. First, it can help strike an appropriate balance between various control efforts with respect to all survey operations. Second, it enables us to dissect the coding error in a given coding operation.

The emphasis in the studies presented in this thesis is in general not explicitly directed towards the measurement of coding errors as they appear in the MSE-relation. Instead, in order to improve coding procedures, we have tried to conduct studies aiming at identifying inaccurate coding procedures, finding efficient verification procedures, and illuminating error structures.

6.3 Schemes for statistical quality control

Manual coding can be characterized as an endless sequence of operations, and it thus seems rather well suited for the application of statistical quality control schemes as originally developed for industrial applications. More specifically, control of coding could be based on various quality control sampling plans. However, coding differs somewhat from, say, car manufacturing. Often there is a problem in finding the true code number, and this therefore forms a sharp contrast to the situation where the diameter of a screw nut is to be checked. As a consequence, errors of the first and second kind are usually much more common in administrative applications than in other quality control areas. Furthermore, it is often impractical or even impossible to establish risk functions for producers and consumers, since coding is only one part of the statistical production process. Nevertheless statistical quality control has been used for several decades as a means for keeping the desired quality level of coding.

The desired quality level and the available financial resources determine the extent of verification. The choice must be made between different sampling schemes (including the special case when all elements are verified). Verification can be carried out on a total or a sampling basis; a combination of those approaches is common.

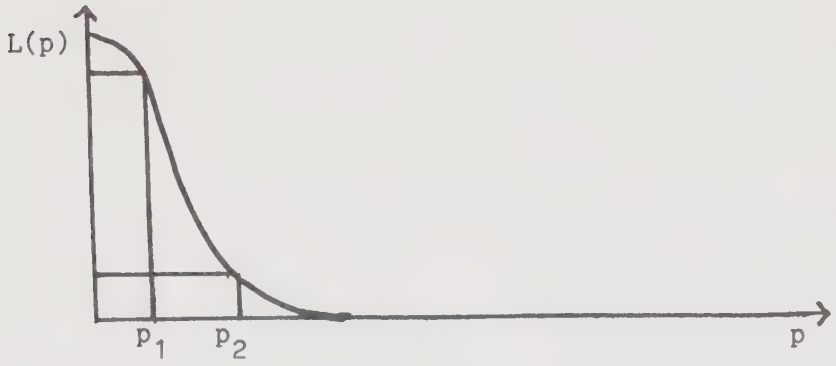
Total verification means that each element in a batch or a lot is verified. The procedure is costly and time-consuming and is thus seldom used throughout an entire manual coding process. The usual approach is to start with total verification, continue until some kind of control of the process is achieved, and then change to sampling verification combined with rectifying inspection of substandard lots.

A sampling inspection plan can assure quality of any prespecified level. The literature discussing such plans is extensive; an early example is Dodge and Romig (1944).

The statistical quality control of coding aims at controlling the code numbers assigned (a control oriented towards the user of data) and at controlling the coder (a control oriented towards the producer of data). The major instruments available are acceptance sampling, process control, and combined procedures which utilize both acceptance sampling and process control. Applications of these techniques may be found in Fasteau et al. (1964), Minton (1969), US Bureau of the Census (1965) and Minton (1970 b).

In order to protect the users of data, acceptance sampling schemes can be applied. In acceptance sampling it is determined whether or not a coded lot of elements should be accepted. A decision is reached for each lot by means of sampling verification. If the lot is unacceptable it is subjected to rectifying inspection, i.e. total reverification. Empirical evidence shows that the coding error rate often decreases with time. Thus at the beginning of a manual coding operation there is no use applying a sampling scheme: too many of the lots would have to be rectified, at too great expense. That is why most manual coding productions can benefit from an initial stage of prequalification including total verification.

A sampling scheme aiming at establishing lot quality typically includes an acceptance number c. The scheme is defined by this number together with the sample size, n. The sampling scheme is often characterized by its operating characteristic curve (OC) and its average outgoing quality curve (AOQ). The OC-curve represents the probability L(p) of accepting a lot with a given error rate. A common OC-curve looks like this:



The error rate p_1 is sometimes called the acceptable quality level (AOQ) and p_2 is called the lot tolerance percent defective (LTPD). AOQ, on the other hand, is defined as the lot average quality after control, assuming a certain quality prior to control. The maximum value for AOQ (AOQL (Average Outgoing Quality Limit)) can be used as the guarantee of the sampling plan.

Process control establishes the sample designs for measuring the quality properties of the process. Furthermore it includes rules for adjusting the process when it is out of statistical control.

A common approach is to use a combination or hybrid of acceptance sampling and process control when dealing with administrative applications such as coding.

6.4 Specific coding control schemes

There are certain control schemes designed specifically for coding. These schemes are applicable in three different areas, namely

- Training of coders
- Dependent and independent verification
- Evaluation

6.4.1 Training of coders

The training and education of coders is indeed valuable since the error rate often decreases with time. If it is possible to "cut" error rates at the beginning of a coding process one will probably end up with a higher average outgoing quality.

Literature on the training and education of clerks is not very extensive. However, the subject is discussed in Minton (1969) and in Dalenius and Frank (1968). In the latter reference, the idea of using master sets is presented. A master set is a set of elements for which the correct coding is known, and could be used during the training period but also as a device for controlling the production process.

6.4.2 Verification

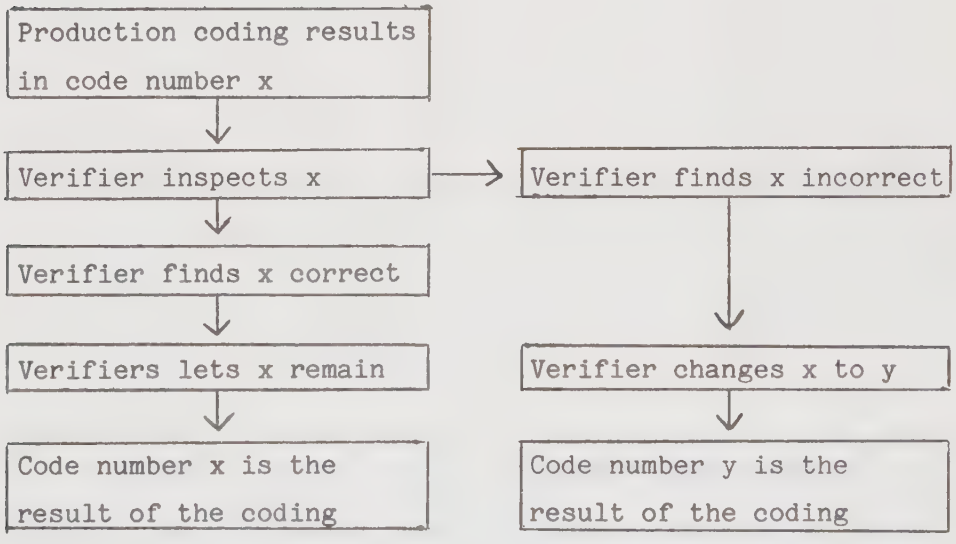
There are two main schemes for verification of coding, i.e. for deciding whether or not a code number is correctly assigned: dependent and independent verification. In dependent verification the verifier has access to the code number assigned by the production coder. In independent verification the verifier has no such access and the decision on outgoing code number must be based on different rules such as majority or modal rules. Within these major schemes several realistic sub-schemes can be defined. Dependent and independent verification is dealt with in Lyberg (n.d.), Lyberg (1969) and Minton (1969).

Let us start with a definition of dependent verification.

> An element is coded by production coder A. The code number is then reviewed by verifier B. B inspects the code number assigned by A and decides if it is correct or not. If it is considered correct it remains unchanged; otherwise B changes the code number.

With this type of verification experience tells us that the verifier has a tendency to let erroneous code numbers remain unchanged: his judgment is influenced by (depends on) the code number assigned by A. Various studies show that the proportion of incorrect code numbers which remain unchanged can be substantial. At Statistics Sweden we often use 50 % as a rule of thumb. As will be seen later, this is not a pessimistic figure. The notion of dependent verification can be extended to include two or more verifiers working either in a group or individually.

Dependent verification can be illustrated by the following scheme using a single verifier.


Production coding results in code number x
Verifier inspects x
Verifier finds x incorrect
Verifier finds x correct
Verifiers lets x remain
Verifier changes x to y
Code number x is the result of the coding
Code number y is the result of the coding

A scheme such as the one just described may prove rather inefficient.

Since the disadvantages of dependent verification can be traced to the element of dependence, the obvious option is some kind of independent verification. This latter alternative has been used, for instance, in the 1960 and 1970 US Censuses of Population and Housing and in the 1970 and 1975 Swedish Censuses of Population and Housing.

Independent verification is defined as follows:

> An element is production-coded by a coder A. The code number is denoted x_A. The element is also coded by N other coders, where $N \geq 1$. Their code numbers are denoted x_B, x_C, ..., x_{N+1}. Each coder works independently and as a consequence does not know how the others have coded. Thus we end up with a set of code numbers x_A, x_B, ..., x_{N+1}. These code numbers are matched and a decision rule defines the outgoing code number.

This definition gives rise to at least two questions. How do we create a situation in which each coder works independently of the other coders? What decision rules are possible?

The first question is a matter of administrative resources. One option is to code directly on the schedule and then mask the code numbers after each coding. This is a rather clumsy procedure, however, and is seldom used. Another option is to copy the schedules to be verified. This procedure is costly, of course, but it has been used in some studies. A third alternative is to code on special forms. This is the best alternative so far, and could be very smooth in a computerized environment, where the matching is done by a computer program.

There are various possible decision rules. One rather natural one is the majority rule: if a majority of the N+1 coders involved agrees upon a specific code number, then this code number

is the outgoing one. (If a majority is not reached, then special measures are taken.) One early example of this rule is the use of the three-way independent verification system in the coding process of the 1960 US Censuses of Population and Housing. With three coders involved in each decision we have three possible outcomes, which can be denoted 3-0, 2-1 and 1-1-1; a majority is reached in all cases except the last.

A very natural way to improve the efficiency is to use a sequential procedure. In the three-way system this means that we start with two coders. After the matching of their code numbers, it is decided whether a third coder is needed or not. Obviously if the outcome of the first matching is 2-0, then the code number of the third coder is completely unnecessary to reach a majority. His contribution could only lead to either 3-0 or 2-1, and we have probably wasted some money. However, if the outcome is 1-1 the third coder must enter the scene and his contribution leads to a majority 2-1 case or a 1-1-1 case. This sequential system, sometimes called two-way independent system, was used in the 1970 US and 1970 Swedish Censuses of Population and Housing.

Another possible decision rule is the modal rule. If some code number in the set of N+1 code numbers is more frequent than any other code number, then this one is the outgoing code number. This rule has, as far as we know, not been used in production coding, except in the case when N+1=3 where the modal rule is identical to the majority rule.

Thus, independent verification can be carried out in many different ways. It is, almost by definition, a more reliable system than the various types of dependent verification. However, it is costly and time-consuming, and this makes its introduction difficult.

Verification schemes can be administered on a total or on a sampling basis. Several problems occur when using sampling inspections plans in the verification operation, although using such plans is a natural way of solving the budget problem. Often a rectifying scheme is used and it is necessary to let coders flow between total control and sampling control; i.e., it is necessary to control both the product and the coders dealing with it. A great problem in production control is the fact that inspection is not error-free. The impact of these errors on single sampling plans is discussed in Minton (1972). The flow of coders between total and sampling control is another problem. The flow must be regulated by means of some prespecified criterion. In Cook (1959) a special point system is presented in which each coder receives a point for each erroneous coding, and decisions about the fate of the coder are based upon the accumulated number of points received when coding a specified number of schedules. In Minton (1970 a) some other decision rules for administrative applications of quality control are discussed.

6.4.3 Evaluation

Evaluation of coding results provides a basis for the allocation of quality control efforts. We have already given examples of results from different evaluation studies. The results of such studies give suggestions concerning the size and emphasis of the necessary quality control program.

Evaluation studies assume the existence of "true" code numbers, which usually are those generated by more skilled coders or expert coders. By comparing these true code numbers to thqse assigned by the production coders an estimate of the gross error rate for the production coding can be obtained. In addition to the references given in chapter 5, evaluation studies are found in, for instance, Fasteau et al. (1962), Minton (1969), Jabine and Tepping (1973), and US Bureau of the Census (1972 b).

It is obvious that many of the coding errors do not depend on the ability of coders: the codes and the coding manuals may be insufficient and thus cause great variability in the coding process. Improvements of these tools therefore seems an urgent task. Possible actions consist of merging categories, making the code less complex, and making the definitions of the categories closer to reality.

6.5 Automated coding as a control technique

As discussed in chapter 5, manual coding is error-prone, and within- and between-coder variabilities are common. An evaluation of manual coding from the viewpoint of control reveals that the error-detection process is complicated by the difficulty of defining the true code numbers and by difficulties of supervising the coding staff. There are problems in implementing the coding instructions together with all the changes and code supplements which always show up with probability one in coding operations of at least moderate size. Furthermore, variabilities exist at all levels and all times in the process. For instance, a single coder has a tendency to look upon the same coding problem differently at different times. Two or more different coders have an even stronger tendency to look at the same coding problem differently at the same time. Different teams have a tendency to look at the coding problems from different points of view. This pattern applies to experienced coders too.

Generally, manual coding and the necessary control efforts consume lots of time and money. This fact, and the fact that the process generates many errors call for some new solution. A possible alternative is to let a computer take over the process, at least to a large extent.

Computerized or automated coding can be carried out in the following way. The code (in the sense of section 4.2) is stored in the computer. (This code is usually referred to as a "dictionary".) The (verbal) information to be coded is input into the

computer, which is programmed to "match" the input with the dictionary. For those cases with a "match", the corresponding code number is assigned; other cases are referred to manual coding.

The variability in manual coding is not present in computerized coding (except, naturally, for those elements which are referred to manual coding). However, the method presents new problems.

6.6 The problem of design of the control

Errors and variability are not always due to the coders alone. In fact, there is reason to believe that most of the errors and the variation arise from imperfections in organization, coder education, coding instruction and the code itself. According to Juran (1968), similar problems exist in quality control applications outside the administrative field as well.

In large-scale statistical investigations we can think of different control stages when considering such operations as keypunching, clerical editing, and coding. In Minton (1970 a) such stages are called

- non-productive training
- production qualification
- postqualification production

These stages have been used in many of the clerical operations of the last U.S. censuses.

We might also, for reasons of simplicity, talk about

- preventive control
- production control

Preventive control covers a broader area than non-productive training: it includes measures concerning, for instance, the schedule, the code, and the general processing. Production control aims at providing satisfactory output from the production coding, and includes control of the production qualification and the postqualification production.

Preventive control can be divided into four substages. These are

a) Selection and initial education of coders
b) Non-productive training
c) Test coding
d) Various feedback operations

Production control is taken care of by means of various verification measures. Reviews of the control stages are given in Biggs (1970), US Bureau of the Census (1965), and Andersson (1977).

Many problems are encountered in designing the coding control. What good design finally comes down to is proper allocation of available resources to different control measures.

A basic problem is that quality standards are seldom established before the start of a coding operation. Usually the extent of verification and other control measures is constrained simply by the available financial and administrative resources. One reason for this may be the fact that a criterion of the type "at most x % erroneously coded schedules are permitted after verification" is of little value either to the producer or the user of statistics. Coding is only one stage in the statistical production and besides, such a criterion cannot be translated into risk functions. Still, there is a need to develop criteria for the verification and control of coding, since it is important to investigate the effects of errors on statistical tables, on the user's decisions based on those tables, and on other situations in which coded data are used.

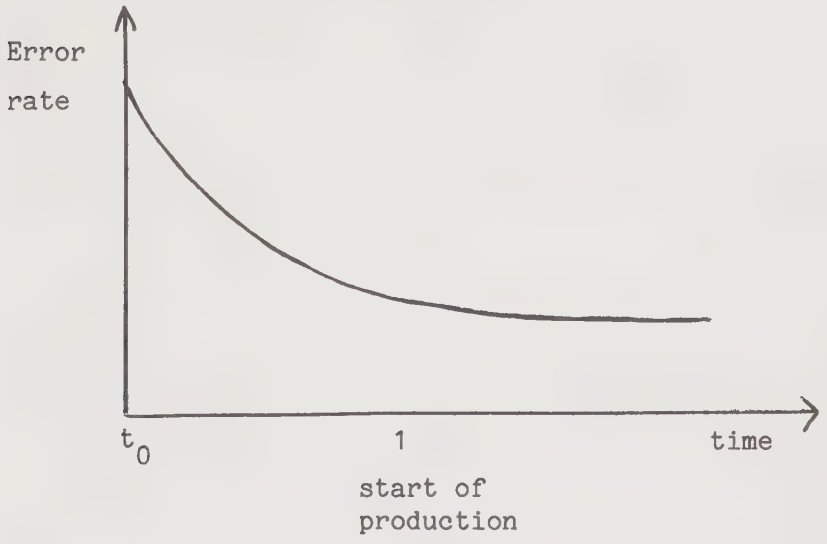
Perhaps experimental studies and pilot studies could serve as a basis for constructing a resource allocation model. However, in large coding operations the pattern of relevant factors is very complex, and a rational allocation of resources relies heavily on support from top management. In census coding there seems to be a cyclic development: failure and success tend to succeed each other on consecutive censuses. The technical facilities change constantly. For instance, in the 1965 Swedish census coding was carried out directly on the schedules, in 1970 the code numbers were read by means of optical character recognition and in the 1975 census the coders used computer terminals. The 1980 census will utilize computerized coding. Other drawbacks, from a resource allocation point of view, are that many of the staff are new in each census, new variables are added and some other are excluded. Some political decisions may have an immense impact on planning. Thus, it was once suggested that the 1980 Swedish census should be carried out by combining information in administrative records with data collected directly from the population.

With these problems in mind, it is understandable that management does not bother too much about general allocation models. However, in specific parts of the total coding program allocation issues can be discussed fruitfully.

Let us assume that the error rate per coder decreases as shown in the following figure:

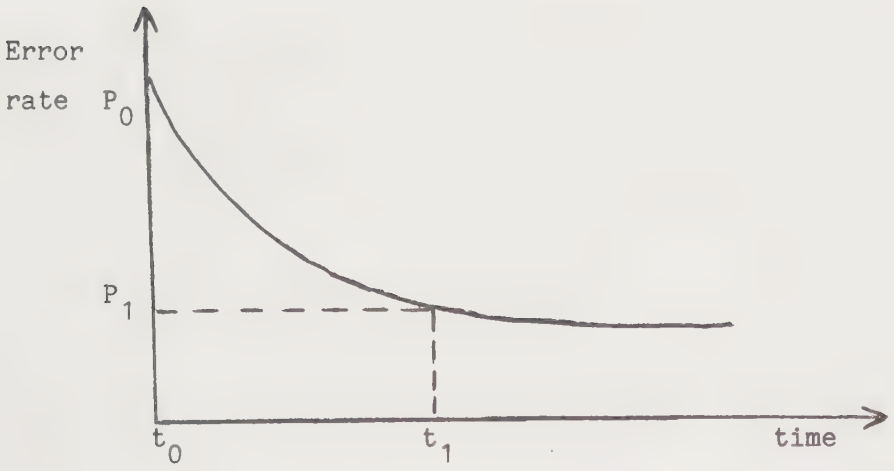
Start of production

In order to prevent the statistical results from being too loaded with errors a realistic approach seems to be to cut down the initial high error rate in one way or another, so that the following more pleasant curve is produced.



In the Swedish censuses the problem of initial low quality has so far been attacked through carrying out the initial control by means of total verification. When the error rate curve flattened out, total verification was replaced by sampling inspection. However, when using rather inefficient control methods like dependent verification this is not a good solution. It can be shown that acceptance sampling verification from the start is a less costly alternative if the administrative costs for rejected lots are within specified limits. The reason that total verification is used is that one expects only very few of the lots to be accepted during the initial stage: thus total verification is used, rather than sampling inspection with almost total rejection. But the fact remains that total control is not a good solution to the problem of high initial error rates.

The peak of this curve could instead be cut by means of a preventive control program, for example, the use of master sets. Of course, cutting error rates by test-coding non-production material costs a lot of money. Let us look at the figure once more.



Although the figure is extremely simplified, we can see that the question is whether it is possible to cut the initial error rate P_0 down to a stabilized production level P_1 before production really begins. If there is a method for reducing P_0 (for instance, master set test coding), it is obviously profitable to carry it out if the following holds:

Master set costs + production cost, given error rate P_1 from the start < production cost, given error rate P_x during time (t_1-t_0) + production cost, given error rate P_1 from time t_1.

Another kind of allocation problem is distinguishing between significant variables and less significant variables. A successful screening of this sort can lead to a more efficient allocation of verification resources.

In addition, many practical problems are bound to arise. For instance, manual coding at an agency often must utilize some kind of coding team organization. A question with no cut-and-dried answer is: How should these teams be organized in terms of number of coders, skillness of different coders etc., so that the best result is obtained?

Other problems are related to dealing with bad coder performance.

In large-scale surveys coders are often hired on a temporary basis. The results of their training are used to decide whether they can be used in production or not. US Bureau of the Census

(1974) describes some experiments concerning the relationship between test scores and coding ability. In general this relationship is weak, and reliable screening of potentially poor coders does not seem possible. At the US Bureau of the Census coders who do not meet quality standards are removed from the operation and a certain reduction of the coder population is calculated in advance. Thus in the plans for the coding operations in the 1970 US Census of Population it was stated that about 16-17 percent of the coders would be removed from the operation sooner or later (Jabine (1970)).

This system is, for various reasons, impossible to apply in Sweden. Coders who are unable to produce according to standard are retrained or possibly transferred to another post. From a strict quality-control viewpoint, this situation limits the possibilities of obtaining satisfactory results.

Part II MANUAL CODING

Contents

	7	Introduction to part II
	8	The importance of "good input"
A		PREVENTIVE CONTROL
	9	More on the design of the code
	10	Selection and training of production coders and verifiers
B		PRODUCTION CONTROL
	11	Dependent verification
	12	Independent verification
		Appendix I
		Appendix II
	13	The extent of verification
		Appendix III
C		SOME ADDITIONAL TOPICS
	14	Errors in coding: detailed discussion
	15	Prior information when designing a manual coding process
	16	Decentralized coding
		Appendix IV

7 Introduction to part II

Section A, PREVENTIVE CONTROL, consists of two chapters dealing with the design of the code and the selection and training of coders and verifiers. Section B, PRODUCTION CONTROL, discusses two basic problems: what kind of verification should be used and how much? Two general verification procedures, dependent and independent, are presented in detail. In that context several empirical results are given. Section C, SOME ADDITIONAL TOPICS, gives a detailed discussion of coding errors as well as some examples of using prior information when designing a manual coding process. It also discusses decentralized coding, i.e. coding carried out by the respondents, the interviewers or by some staff outside a central agency.

8 The importance of "good input"

One of the axioms in quality control is that you cannot inspect quality into a material: the quality must be there from the start. If production coding is of substandard quality prior to verification and control, then the costs of verification and control will be excessive. Let us illustrate the effects of "good input" with a few examples.

8.1 Choice of code

If it is possible for the designer of the coding operation to influence on the construction of the code, he should certainly take this opportunity, since he can thus improve the outgoing quality. During the 1970 Swedish census of population it was found that some verbal descriptions of occupation tended to be erroneously coded because that the code was so detailed that it did not fit some of the rather vague descriptions given. As a consequence, in the 1975 census some categories of occupation were merged into broader categories. Examples of such merging were

i) chemists + physicists
ii) tailors + dressmakers
iii) housepainters + lacquering men
iv) telephone operators working at telephone company + telephone operators working at offices

The code changes resulted in less general vagueness and as a consequence we achieved a smoother control operation. Of course this was achieved with the loss of information about the previous, more narrow, categories. However, since these were considered unrealistic from a practical point of view, we could certainly live with that loss.

8.2 Verification of "good input"

It is typical of dependent verification that the verifier does not identify all errors made by the production coders in the inspected material; the proportion of errors identified will depend on the skill of the verifiers, as does of course, the proportion of errors made by the production coder . This actualizes an important question: who is to serve as production coder and who is to work as verifier? We will respond to this question by means of an example.

Thus consider a situation with two categories of coders; the number of coders is the same in both categories. One category codes with a 10 % error rate and the other with 20 %.

Let us assume that - irrespective of coder category - the proportion of errors identified in the course of the verification is stable around the value $\Pi = 50$ %. The usual system is that the "20 %-category" is used as production coders and the "10 %-category" as verifiers. If all the material is verified the outgoing error rate will be $20 \cdot 50 \cdot 10^{-2}$ % or 10 % with the usual system. If instead the "10 %-category" is used as production coders and the "20 %-category" is used as verifiers, the outgoing error rate will be $10 \cdot 50 \cdot 10^{-2}$ % or 5 %. The example will illuminate the idea. Of course this reverse system does not entirely depend on the fact that Π is stable. The two coder categories may have different Π-values, Π_{10} and Π_{20}. The main thing is that the relation

$$10 \cdot \Pi_{20} < 20 \cdot \Pi_{10}$$

holds.

Despite what has been said about Π-stability with different verifiers, Π may depend on the error rate before verification (p). A closer analysis of the drawbacks of dependent verification suggests that Π may be positively correlated with the error rate. In that case the resulting relation becomes

$$10 \cdot \Pi_{20|p=10} < 20\ \Pi_{10|p=20}$$

Also, when we use independent verification we are much better off if the input is good, since if it is a decision on outgoing code is reached faster than if it is not.

Using the best coders as production coders was suggested in planning both the 1965 and the 1970 Swedish censuses of population and housing but there seems to be a psychological resistance among management for dealing with the error problem in such a way. The evaluation studies show that Π in fact is rather stable, so probably such a system is very useful.

8.3 "Good input" in statistical quality control

The efficient choice of an acceptance sampling plan assumes that the error rate before verification, p, is known in advance. If p is high and the desired quality is lower than p, statistical quality control simply does not work: too much rectifying inspection is needed and the result might even be that total verification is a cheaper alternative.

A PREVENTIVE CONTROL

9 More on the design of the code

Often the users of survey data want a very detailed coding which is difficult if not impossible to implement given a specific response pattern of verbal descriptions. In Sweden, at least, there seems to be little if any connection between the empirical verbal response pattern and the code construction. The result is coding variability. In Denmark (see Skotte 1977)a system is used in which respondent descriptions of their occupations are the major constituent elements when building the code for that variable. Besides the possible lack of coordination between response pattern and code construction, there do not seem to exist any national or international recommendations for this type of work. Of course, much has been written about codes and the construction of categories, but where does one find rules for enlarging a given code and where can one find recommendations aimed at achieving a smooth production process? Some of these problems are further discussed in Felme (1978, 1980) and in Felme et al. (1977).

The construction of the code should include a consideration of at least the following aspects.

The general desiderata should be listed and evaluated simultaneously by the users and producers of the statistics. The desiderata should be weighed against the prevailing response pattern. Sometimes a formal definition of a category gives us trouble: for instance, the user might be interested in a certain partitioning of a broader class of occupations into more narrow categories. However, the response pattern might be so standardized that a coding according to the user's special wishes is impossible, in that to achieve a correct coding the coder would have to rely on auxiliary information which perhaps is diffuse or simply does not exist. This difficulty seems to be one of the major, if not the major, sources of coding

errors. From a dogmatic point of view this kind of error is easily removed: a much better coding can be obtained if special "trash categories" for "non-specified" cases are used whenever there is any doubt about the coding of a specific description. However, we cannot ignore the user's needs and handle in this manner all the way. The result would be statistical tables of minor value. Also, the cell frequencies for the "trash categories" would be relatively high, and the result of such a coding would be that we would have in effect a non-response situation. Of course, one solution is to use, in the first place, a better and more costly measurement procedure that makes more accurate information possible, but then we have to make another survey.

Many of the errors in the 1970 Swedish census coding could have been avoided if code numbers for "non-specified" cases had been used more often. For instance, every major occupation and industry group include a "trash category". These categories were, however, not always used in cases of vagueness, since the administrators were afraid of the "non-specified" problem within major groups. Thus the "non-specified" problem was traded against a coding problem. When designing the 1975 census it was decided that the "trash categories" should be used more frequently. In order to avoid the "non-specified" problem, or at least reduce it, the regular coding was followed by a special procedure in which complete information about vague cases was gathered on a sampling basis. These two sources of information were combined, and this resulted in a more valid estimate of minor group totals. The procedure is described in Andersson (1974).

In an evaluation study of the Swedish Labor Force Survey the main causes of errors were noted for each of the erroneous industry code numbers discovered in the sample. It is quite clear that most of the errors were due to the code or to the

means used to apply the code. Only about 24 % of the coding errors could be totally attributed to the coder. This information has lead to several changes and clarifications in the instructions for coders and interviewers.

Thus the vital points are that "trash categories" should be used more often and that the respondent patterns should be carefully analyzed with a view to collapsing categories on the list of what the users would like. A relatively small number of such amalgamations can reduce the error rate substantially.

10 Selection and training of production coders and verifiers

10.1 The concept of a master set

The selection of coders and verifiers should be based on coding performance. Information on coding performance is arrived at after the necessary initial education and training period. Such non-production training and test coding both can be carried out by means of a master set. The concept of a master set is fully discussed in Dalenius and Frank (1968). The main feature of such a set is that it is known to which category each element in the set belongs to.

By using the master set technique the categories themselves can be tested and ill-defined categories can be properly redefined. Another result possible is the screening of coders into different groups - for instance, coders ready for production coding and coders who need more on-the-spot training. Thus non-production training aims at

i) correcting coder errors and discussing each error with the coder in order to upgrade performance

ii) deciding when coders have achieved the performance standards for production coding.

Discussing errors with the coders is a typical feedback operation. The identification of error-prone categories can result in changes in the code and/or the coding instructions.

Besides making it possible to identify good coders, the master set can be used during an ongoing production to investigate whether or not the coders' performance is deteriorating.

Such a training technique can be run in parallel with the ordinary production and its control system. For instance, the ordinary production control system may not be quite adequate to identify bad coders; in this case we can talk about preventive control too.

As pointed out in Dalenius and Frank (1968) and Minton (1968), there are certain practical problems connected with the construction and the use of a master set. For instance, the elements of the set might be chosen from the ongoing production, from certain pilot surveys or from an earlier statistical study of the same general kind as the ongoing production. The last approach seems the most practical, but on the other hand this choice brings with it certain limitations from the user's standpoint. Among other things, it could not serve as a device for process control. However, for the parallel system mentioned above it could serve well.

Another problem is whether to use a "large" set or several "smaller" subsets, where these smaller subsets differ in coding difficulty or where some elements may be common to several sets to provide information about the within - or between - coder variability.

In the 1970 Swedish Census of Population a master set was constructed to provide a means for efficient preventive control. The elements in that master set consisted of verbal descriptions from 1000 schedules from the 1965 census. (Time did not permit us to use material from the 1970 census.) These descriptions were manually mapped on a set of 1970 census schedules in order not to confuse the test coders. Some variables of course were not common to both censuses but the main variables were. Thus the master set consisted of partly realistic and partly fictitious information. The fictitious information was generated by coders with much experience in census coding. The 1000 master set schedules were a subjective sample. The following criteria were used at the sampling phase:

a) Since the industry, education and occupation structures differ between different areas in the country, a simple stratification was called for. The strata were pure urban areas, pure rural areas, and other areas.

b) It is possible to distinguish, in a subjective way, between "easily" coded schedules, "average" schedules, and "difficult" schedules. It was decided that the proportion between these types should be 2:3:5 in the set, i.e. that the proportion of difficult cases should be higher than in regular census material.

The elements of the master set were coded with the constraint that for each variable each element should have a "true" code number associated with it. Each element was independently coded by two experts. After that the two sets of assigned code numbers were matched for each element. If there were no differences the set of code numbers was accepted as "true". If at least one disagreement occurred the differing code numbers were reconciled and true code numbers were assigned. If a true code number for a variable could not be established the element was removed from the master set. The final set of true code numbers for the elements was written down on schedules for optical character recognition, and these schedules were read and stored in the computer. This routine permitted automated comparison between the code numbers assigned by the coders and the true code numbers of the master set.

Unfortunately, the automated matching did not work because of the initial bad performance of the optical character recognition system. The matching had to be done manually and as a consequence we obtained no reliable statistics on coder 'performance. Let us just give a small example of error rates.

One of the subsets was coded by 31 coders. Taking all variables into consideration, the percentage of elements with at least one coding error ranged from 23.9 % to 46.7 %. More specifically, the overall error rate for the variable industry was 20.3 % and for occupation 20.5 %. However, there are reasons to believe that the coders managed to code only the simplest cases, since for 18.9 % to 47.0% of the elements in the subset, the coders opted not to code at all.

We were happy that this coding material was a master set instead of production material.

From an educational point of view, however, the master set worked well and it was used during two weeks before the coders were put on production work.

10.2 A further comment

In Sweden we have found that the master set is a good training device. Most of the coding at Statistics Sweden is now carried out by a permanent staff. Some years ago new labor market legislation came into effect and bad coding performance became no longer a valid reason for dismissal of coders. Thus the selection of coders presents no serious problems since we have to use what we have. We have of course specific selection problems of the type mentioned in chapter 8. The best coders should be used as production coders rather than as verifiers.

In US Bureau of the Census (1974) the concept of learning curves is treated, and learning curves for production and quality are determined. These curves can serve as a model of the actual learning process; information such as they present could be useful when the amount of coding education necessary is determined.

B PRODUCTION CONTROL

Production control presents two basic problems: the kind and the amount of verification to be used. In this section the two basic verification procedures, dependent and independent verification, are presented in detail. Finally we discuss the question of how much verification, if any, should be carried out.

11 Dependent verification

As stated in chapter 6, in dependent verification the verifier has access to the code number to be verified. This is the very drawback of the approach. In this chapter the performance of dependent verification is illustrated by some experimental results.

11.1 An evaluation of dependent verification

In 1967 a study, presented in Lyberg (n.d.), was conducted in order to illuminate, in a concrete fashion, the performance of the dependent verification used in the 1965 Swedish Census of Population. A general evaluation of the coding was obtained as a byproduct.

Until that time the use of dependent verification had been more or less taken for granted. For instance, in the design of the coding control operations of the 1965 Census, no alternatives were proposed. However, evaluation results from the US Bureau of the Census eventually reached us and the dependent verification system was questioned. Before we go on we will give a short review of the US experiences that led to this situation.

Dependent verification was used for the coding in the 1950 US Censuses of Population and Housing. The control scheme is presented in US Bureau of the Census (1956). By means of error

noting and error planting it was found later on, when the 1960 Census was planned, that the dependent system was inefficient. The effectiveness of the dependent system is indicated in Table 11.1, taken from Fasteau et al. (1964).

Table 11.1 Percent of errors not detected by dependent verifiers in verification of occupation and industry and general coding in the 1950 Census of Population (one month's experience, November 1950).

Coding process	Operations office and basis of measure	
	Washington DC (Planted errors)	Philadelphia (Noted errors)
General coding	29	47
Occupation and industry coding	43	69

This result together with other experiences called for a new verification system, and thus in the 1960 Census of Population the three-way independent verification system appeared on the scene. Dependent verification was used only during the coder qualification period and its drawbacks persisted. (In a more recent experiment, described in US Bureau of the Census (1977), dependent verification missed about two-thirds of the errors when the coding of industry and occupation was controlled.)

In our Swedish study verified schedules from the 1965 census material were sampled from four strata covering about 70 percent of the population. The stratification was based on geographic area and on whether the schedules were verified totally or on a sampling basis. The study was confined to the following four variables:

(1) Relationship to head of household (1-digit)
(2) Employment (1-digit)
(3) Status (1-digit)
(4) Industry (3-digit)

The samples were coded by a team of three experimental coders who were considered especially skilled. Each coder coded independently of the others and then the code numbers were matched. Three cases were possible. First, all three coders could agree; we call that case 3-0. Second, two could agree but not the third; we call that case 2-1. Finally, we have the case when no two coders agree; we call that case 1-1-1. In the first and second cases, clearly, we were able to define a majority code number; that number was used as an evaluation code number. In the 1-1-1 cases we let an expert decide the evaluation code number. After that the evaluation code number was compared both to the unverified production code number (P) and to the dependent verification code number (V). Table 11.2 shows the results for variable (1) for one of the subsamples.

Table 11.2 a A comparison between the majority code number and the production code number (P_1) (number of cases).

Variable 1, Subsample 1

Experimental coder combinations	P_1 agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	1131	-	-	16	-	1147
2 - 1	-	2	3	0	-	5
1 - 1 - 1	-	-	-	-	0	0
						1152

Table 11.2 b A comparison between the majority code number and the dependent verification code number (V_1) (number of cases)

Variable 1, Subsample 1

Experimental coder combinations	V_1 agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	1141	-	-	6	-	1147
2 - 1	-	2	3	0	-	5
1 - 1 - 1	-	-	-	-	0	0
						1152

Obviously the variable is an easy one to code: among the experimental coders only 5 out of 1152 cases caused some kind of disagreement. The error rate among production coders is estimated by summing the number of deviations from the majority code number, i.e. the frequencies inside the triangle. Thus the estimate of the error rate in this case is 19/1147 or 1.6 per cent.

The 19 cases within the triangle were reduced to 9 by means of the verification system and the system handled an estimated 53 per cent of the errors.

Now let us turn to the more complex three-digit variable (4). The corresponding results are given in Table 11.3 below.

Table 11.3 a A comparison between the majority code number and the production code number (P_4) (number of cases).

Variable 4, Subsample 1

Experimental coder combinations	P_4 agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	427	-	-	48	-	475
2 - 1	-	41	24	8	-	73
1 - 1 - 1	-	-	-	-	5	5
						553

Table 11.3 b A comparison between the majority code number and the dependent verification code number (V_4) (number of cases).

Variable 4, Subsample 1

Experimental coder combinations	V_4 agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	451	-	-	24	-	475
2 - 1	-	44	23	6	-	73
1 - 1 - 1	-	-	-	-	5	5
						553

This is a much more difficult variable for the coders: they agreed in only 475 of the 553 cases or in 86 per cent. The error rate in production coding, estimated by (48 + 8 + 24)/548, is a striking 14.6 percent.

The 80 cases within the triangle in Table 11.3 a were reduced to 53 by means of the verification system. This time the system took care of only 34 percent of the errors.

A summary of the error reduction rates for all variables and subsamples in the study is given in Table 11.4 a below.

Table 11.4 a Estimates of error reduction rates (%) when using dependent verification (rounded figures). (Absolute error rate before verification within brackets.)

Subsample \ Variable	1	2	3	4
1	53(19)	16(31)	18(11)	34(80)
2	38(13)	7(15)	33(6)	11(37)
3	32(19)	45(38)	6(17)	30(74)
4	0(9)	41(17)	25(16)	19(52)

Clearly, this type of calculation needs some further elaboration. First, the choice of the majority code number for evaluation purposes certainly has its drawbacks: the 2-1 cases are an indication of variability in the process. To charge a coder with an error when he has assigned a code number that agrees with the one assigned by one of the experimental coders is a dubious procedure. In the original study these doubts were handled so that the comparison between production code number or verification code number with the majority code number led to a special classification of disagreements. If the code numbers P or V agreed with none of the experimental coders in the 3-0 case, this was called a "very serious" disagreement. If P or V agreed with none of the experimental coders in the 2-1 case, this was called a "serious" disagreement. If P and V agreed with one of the experimental coders in the 2-1 case, this was called a "less serious" disagreement.

As a result using this operational rule might lead to an overestimation of the overall error rate per variable. Furthermore, since the coder is charged with errors resulting from "less serious" disagreements, we probably get an exaggerated picture of the ineffectiveness of dependent verification. In table 11.4 b we show estimates of error reduction rates for "very serious" and "serious" disagreements only ("softened" definition of error).

Table 11.4 b Estimates of error reduction rates (%) for dependent verification with a softened definition of error (rounded figures). (Absolute error rate before verification within brackets.)

Subsample \ Variable	1	2	3	4
1	63(16)	12(25)	13(8)	46(56)
2	50(10)	11(9)	67(3)	19(27)
3	25(12)	62(29)	20(10)	42(52)
4	-14(7)	47(15)	42(12)	29(38)

The reduction rates are in general higher with the softened error notion. Let us finally find out what will happen when we are considering "very serious" disagreements only: see Table 11.4 c.

Table 11.4 c Estimates of error reduction rates (%) when using dependent verification with a further softened definition of error (rounded figures). (Absolute error rate before verification within brackets.)

Subsample \ Variable	1	2	3	4
1	63(16)	12(25)	13(8)	50(48)
2	50(10)	11(9)	67(3)	10(20)
3	33(9)	62(29)	20(10)	47(45)
4	-14(7)	50(14)	42(12)	33(33)

Little changes when we omit "serious" disagreements since the number of such cases is small.

The estimates of error reduction rates suffer from small sample sizes generated by low or very low absolute error frequencies. As a result the confidence intervals for reduction rates are very wide. The variation between subsamples is considerable, in part because of the small sample sizes and the sensitivity of the percentage measure and in part because of real differences between different substrata.

However, the most important observation is that the estimated error reduction frequencies do not come close to the ideal 100 %.

Considering that the cost of the verification operation during the 1965 Census coding was more than 20 per cent of primary coding cost, some might argue that the whole verification operation was a waste of financial resources. (Of course this is

not true: the very existence of a quality control program has a salutary psychological effect on all persons involved in the coding operation.)

For the industry variable we took a closer look at the gross and net effects of the dependent verification system. Consider Table 11.5.

Table 11.5 a-b. The gross and net effects of dependent verification: the majority code M_4 compared to P_4 and V_4.

Table 11.5 a Subsample 1
The case 3-0

	$P_4=V_4$	$P_4\neq V_4$	Total	Effects
$M_4=V_4$ $M_4=P_4$	426	-	426	None
$M_4\neq V_4$ $M_4=P_4$	-	1	1	Destructive
$M_4=V_4$ $M_4\neq P_4$	-	25	25	Constructive
$M_4\neq V_4$ $M_4\neq P_4$	23	0	23	None
			475	

Table 11.5 b

The case 2-1

	$P_4=V_4$	$P_4\neq V_4$	Total	Effects
$M_4=V_4$ $M_4=P_4$	39	-	39	None
$M_4\neq V_4$ $M_4=P_4$	-	2	2	Destructive
$M_4=V_4$ $M_4\neq P_4$	-	5	5	Constructive
$M_4\neq V_4$ $M_4\neq P_4$	25	2	27	None
			73	

For the case 3-0 we find that 26 changes of the production code number have been made by the verifier: 25 of these are constructive changes while 1 is destructive. All five coders agree in 426 cases. In 23 cases the verifier has accepted a production code number differing from the majority code number.

The 2-1 case is in itself an indication of a difficult coding situation. The verifier seldom changes an erroneous or doubtful code number: he has a tendency to give a "verdict of acquittal". However, the tables show that at least the destructive changes are quite few.

Generally, the same pattern is found in the tables for the other three subsamples.

In order to study how the experimental coders performed when they acted as dependent verifiers, we asked them to code one-third of the entire study material with access to the production code number. This special study was carried out before and

after the main study: thus for two subsamples (each consisting of about one sixth of the total material), the following sets of observations were made available for, say, variable 4:

X_{4d1} and X_{4d2}
Y_{4d1} and Y_{4d2}
Z_{4d1} and Z_{4d2}

where X, Y and Z denote the different experimental coders and 1 stands for before and 2 stands for after. The figures in Tables 11.6-11.8 below come from parts of subsample 1.

Table 11.6 a-b The performance of experimental coders acting as dependent verifiers: M_4 compared to X_{4d1} and X_{4d2} respectively.

Table 11.6 a

Experimental coder combinations	X_{4d1} agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	83	-	-	3	-	86
2 - 1	-	10	3	1	-	14
1 - 1 - 1	-	-	-	-	1	1
						101

Tabell 11.6 b

Experimental coder combinations	X_{4d2} agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	65	-	-	1	-	66
2 - 1	-	3	7	2	-	12
1 - 1 - 1	-	-	-	-	1	1
						79

Table 11.7 a-b The performance of experimental coders acting as dependent verifiers: M_4 compared to Y_{4d1} and Y_{4d2} respectively.

Table 11.7 a

Experimental coder combinations	Y_{4d1} agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	83	-	-	3	-	86
2 - 1	-	8	5	1	-	14
1 - 1 - 1	-	-	-	-	1	1
						101

Table 11.7 b

Experimental coder combinations	Y_{4d2} agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	62	-	-	4	-	66
2 - 1	-	4	5	3	-	12
1 - 1 - 1	-	-	-	-	1	1
						79

Table 11.8 a-b The performance of experimental coders acting as dependent verifiers: M_4 compared to Z_{4d1} and Z_{4d2} respectively.

Table 11.8 a

Experimental coder combinations	Z_{4d1} agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	84	-	-	2	-	86
2 - 1	-	8	5	1	-	14
1 - 1 - 1	-	-	-	-	1	1
						101

Table 11.8 b

Experimental coder combinations	Z_{4d2} agrees with ... experimental coders					
	3	2	1	0	Expert cases	Total
3 - 0	66	-	-	0	-	66
2 - 1	-	9	3	0	-	12
1 - 1 - 1	-	-	-	-	1	1
						79

Comparison between the results of the individual experimental coders and the majority code number M is complicated by the fact that the latter is a consensus of the former. The triangle we used before to isolate the so-called "erroneous" cases might now be replaced by a rectangle enclosing those cases in which an erroneous code number was definitely assigned by the experimental coder acting as a dependent verifier. Unfortunately, given a 2-1 experimental coder combination, we did not study whether an agreement with one experimental coder was an agreement with himself or an agreement with one of the other two. The rectangle used in the tables above isolates the number of cases which are definitely erroneous according to the operational definition of true code number. Thus that number of cases gives us an estimate of the minimum error rate for an experimental coder acting as a dependent verifier. In the tables given above we see that the rectangles contain the frequencies for the different experimental coders shown in Table 11.9.

Table 11.9 Absolute and relative error frequencies for coders X, Y and Z when acting as dependent verifiers (parts of subsample 1).

Error frequency after verification		X	Y	Z
Before main study	Absolute	4	4	3
	Relative (%)	4.0	4.0	3.0
After main study	Absolute	3	7	0
	Relative (%)	3.8	8.9	0
Total	Absolute	7	11	3
	Relative (%)	3.9	6.1	1.7

For the corresponding parts of subsamples 2, 3 and 4 (b_{22}, b_{32} and b_{42} respectively) the results are as given in Table 11.10.

Table 11.10 Relative error frequencies for coders X, Y, and Z when acting as dependent verifiers (subsample b_{22}, b_{32} and b_{42}).

Relative error frequency after verification (%)	X			Y			Z		
	b_{22}	b_{32}	b_{42}	b_{22}	b_{32}	b_{42}	b_{22}	b_{32}	b_{42}
Before main study	4.1	5.0	5.6	5.4	0	0	0	2.5	1.1
After main study	2.9	1.1	0	1.4	2.2	0	4.3	2.2	3.0
Total	3.5	2.9	2.6	3.5	1.2	0	2.1	2.3	2.1

We notice that the experimental coders, in general, produce better results than the census verifiers. But their results are by no means error-free. The whole study is described in Lyberg (n.d.) and in Dalenius and Lyberg (1968).

We have seen that dependent verification can be ineffective because inspection errors can occur. In some situations dependent verification can be even worse than no verification at all. In Christianson (1972) such situations are described by means of a modification of a model initially developed in Bross (1954). The idea is as follows.

There exists a population of N elements. Each element I is associated with a code number. It is possible to define a variable X_I which takes values 0 or 1 according to whether element I is correctly coded or not. Let Q be the proportion of erroneous code numbers in the population before verification. A dependent verification of this coding results in changes or non-changes of the production code number.

Let us define the variable y_j where $j = 1, 2, \ldots, N$. y_j takes values 1 or 0 depending on whether the production code number is changed or left unchanged by the verifier. Let us further define the two erroneous coding probabilities α and β.

α is the probability that a correct production code number is changed by the verifier;

β is the probability that an erroneous production code number is not changed by the verifier (or changed to another erroneous code number).

The proportion Q_V of erroneous code numbers after verification is a stochastic variable with expectation

$$E[Q_V] = Q\beta + (1-Q)\alpha$$

Obviously, dependent verification pays in terms of quality if the relation

$$Q\beta + (1-Q)\alpha < Q; \quad \text{that is, if } \beta < 1 - \frac{1-Q}{Q}\alpha$$

holds.

In Figure 11.1 below the function

$$\beta_{max} = 1 - \frac{1-Q}{Q}\alpha$$

is graphically represented for some different α-values.

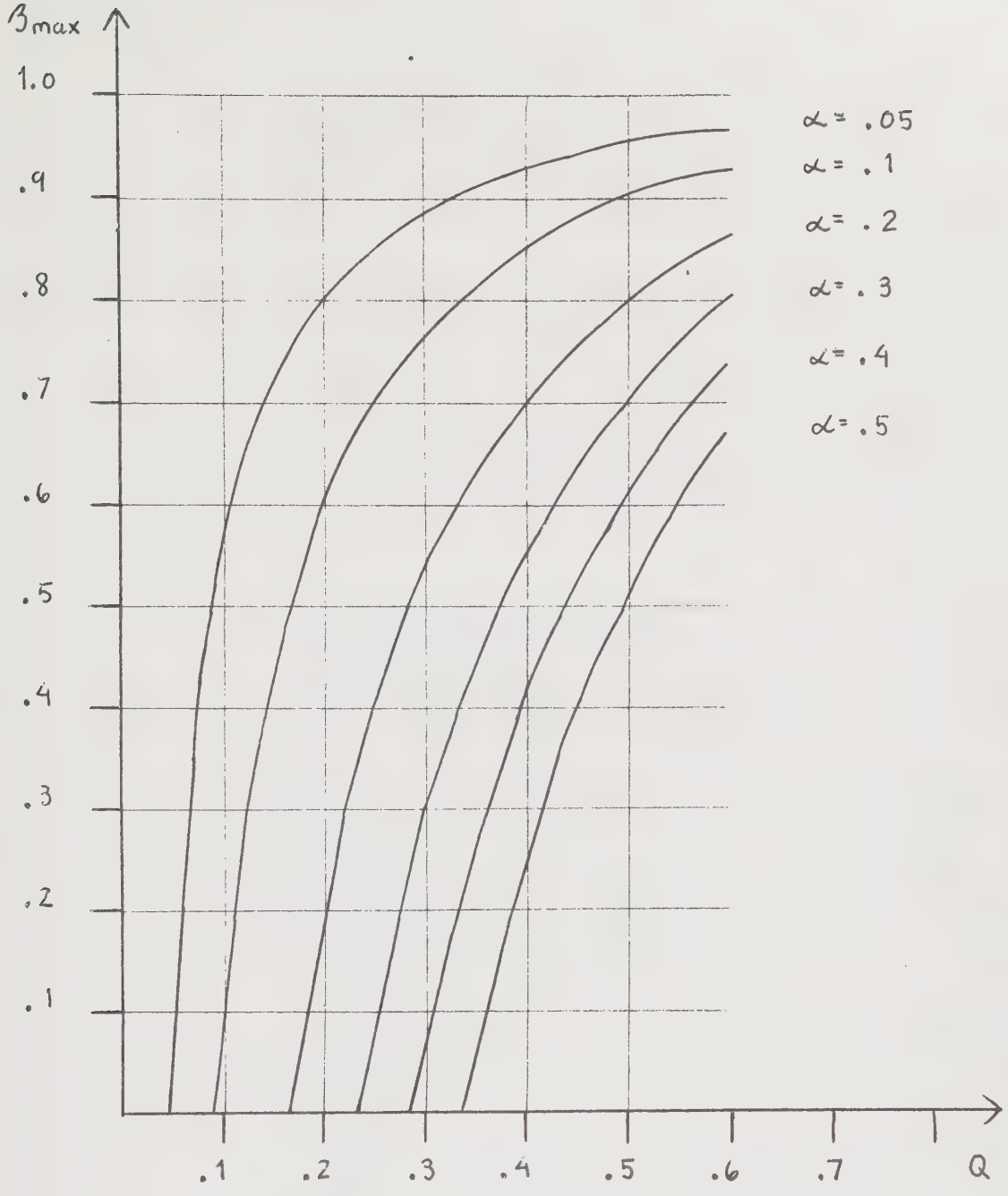


Figure 11.1

Q = the proportion of erroneous code numbers before verification.

α = the probability that a correct code number is changed by the verifier.

β = the probability that an erroneous code number is not changed or changed to another erroneous code number by the verifier.

Dependent verification pays off in terms of quality if and only if $\beta < \beta_{max}$.

Figure 11.1 shows the requirements on α and β which must be fulfilled if dependent verification is to pay. For instance, if $Q = \alpha = .1$ then β must be less than .1; otherwise the quality is worse than with production coding only.

However, as we have seen, empirical evidence indicates that α is often near zero. Thus, most of the time there is no risk applying this kind of verification, though the effects might be just marginal. The proposed analysis implies that Q, α and β can be estimated within rather narrow limits. Evaluation studies of earlier coding operations can be helpful in this case.

It is possible to estimate Q, α and β during an ongoing production even if evaluation data do not exist. In Minton (1969) a list of methods for making these estimates is given. There are in general three types of estimation methods, namely

a) error-planting; here erroneously coded elements are deliberatly planted in the material to be verified. After verification one simply counts how many of the planted erroneous code numbers have been changed by the verifier. The method requires that the verifier be unable to recognize which code numbers are planted. This requirement can be difficult to fulfill since it can be hard to plant realistic errors in realistic proportions.

b) reinspection on a sampling basis; here specially trained coders inspect the production code numbers and decide whether they are correct or not. However, this method is doubtful since it is a dependent method. Its only difference from usual dependent verification is that these verifiers are "experts" and have more time to spend on the job. But even experts have a clear tendency not to correct incorrect code numbers.

c) independent verification; here a sample of elements is coded by one or more verifiers who do not know the production code number. After verification the different code numbers are matched and a decision rule decides the outgoing code number. This code number is compared with the production code number and estimates of Q, α and β can be obtained. The obvious drawback with this method is that the majority is not always right. Furthermore, even if Q can be estimated within rather narrow limits the estimation of α and β often becomes a problem involving sampling of rare items, since these calculations must be based upon relatively small absolute error rates.

As pointed out in Linebarger et al. (1976), dependent verification has certain advantages.The operation is quick, fairly non-disruptive to handle, requires little work if handled clerically, and is rather inexpensive. The serious disadvantages are to be found in the quality dimension. We must have a reasonably low value of our β. The tables in this chapter gives estimates of β which do not favor the dependent system. And if the disadvantages of a verification system are found in the quality dimension we are in big trouble.

These findings do not necessarily apply to all situations. In a recent study on medical mortality coding described in US Department of Health, Education, and Welfare (1980), dependent verification estimated the true error rate rather well. However, this situation occurred in part because of unusually high values of α. The data from that study suggest that dependent verification can produce estimates of error rates which approximate those of different types of independent verification if some basic conditions are fulfilled; namely, that

i) the verification is carried out by the best coders available

ii) the verifiers are sufficiently aware that their work is controlled

iii) the verifiers can work without any time pressure.

However, as the authors point out, at the same time these are some of the conditions one wants to avoid in order to retain the advantages of dependent verification when compared to other verification means.

It should also be pointed out that the use of dependent verification in a sampling inspection scheme could give serious technical drawbacks in at least the two following ways:

i) Errors of the first and second kind cause the operating characteristic curve to shift

ii) Prespecified average outgoing quality limits increase.

These problems will be discussed in chapter 13.

11.2 Some further experiments with dependent verification

There are situations in which dependent verification seems to be the only way to control coded material; for instance, when coding must be carried out in direct connection with the verbal description of the element to be coded. Often alternative solutions, which imply independent verification, are administratively or financially impossible. (However, some administratively successful experiments with invisible ink have been conducted at the US Bureau of the Census quite recently; see US Bureau of the Census (1977).) We have therefore conducted some minor studies in an attempt to produce more efficient dependent verification: we have investigated how dependent verification works in a repetetive situation, and have studied the effects of coding teamwork.

11.2.1 Design of the experiments

Let us first elaborate somewhat on the notions of repeated dependent verification and teamwork.

Repeated dependent verification

A material is production-coded and then goes through three consecutive dependent verifications. Each set of code numbers is compared to the associated set of true code numbers and as a consequence the quality contribution after each run can be measured. According to the definition previously given dependent verification is terminated after the first verification. What would happen when the verifier had access to more than one code number other than his own? We did not know much about that.

Teamwork

The assessment of prior distributions plays an important role in the theory of subjective probabilities. The problem of creating one such distribution out of several is usually called the consensus problem. In psychological experiments it has been shown that a group (for instance a group of experts) often constructs "better" prior distributions than any of the single group members. The reason is that exchange of information takes place within the group.

This idea of a collaboration between people can be applied in a coding situation. Usually we do not permit communication between verifier and production coder at the time of verification because we want unbiased estimates of the error frequencies. Now this might be a two-edged sword. Possibly a better result would be obtained if the coders decided upon a code number together. We also experimented with the so-called Delphi technique in which the coders discuss the codings together within the group but need not expose their own code number publicly:

instead each coder assigns his own code number anonymously after a general discussion. Thus in the consensus study the purpose was to get a final code number which all coders agreed upon. In the Delphi study each element could be assigned different code numbers.

Now let us describe the different experiments.

We had at our disposal material from the coding evaluation of the 1970 Swedish Census of Population. Thus for each element we had access to the "true" code number for the variables industry and occupation. Four regular census coders took part in the experiments. From the evaluation material six subsets (m_1, m_2, ..., m_6) of schedules were sampled randomly. For each schedule the variables industry and occupation were to be coded. It should be noted that the experimental material contained no preassigned code numbers to start with. When we talk about production coding in the subsequent text we mean the production coding as the first step in the experimental coding; perhaps first coding or initial coding would be more proper terms. The comparisons made concern the evaluation code only. The regular census code numbers were never involved in these studies.

Experiment A

Experiment A consisted of three repeated dependent verifications of production-coded material. The quality contribution was measured after each separate verification by comparing the verification code numbers and the production code number with the evaluation code number.

Experiment B

Experiment B entailed group coding by pairs of coders. Each pair discussed the coding openly and the two reached a verdict together. Thus this was a consensus experiment. For one part

of the material the coders had access to the code numbers from experiment A; for the other part they did not. The production coder and the first verifier in experiment A were one of the pairs in this experiment.

Experiment C

Experiment C entailed group coding by all four coders simultaneously. It was carried out in four different ways: consensus with and without code numbers from experiment A (as in experiment B) and Delphi with and without code numbers from experiment A.

Experiment D

In experiment D we carried out parts of experiments A and C in reverse order. This time we used some new materials m_5 and m_6 (which in turn were divided into m_{51}, m_{52}, m_{61} and m_{62}). We conducted consensus and Delphi without giving the coders access to code numbers before repeated dependent verification. The purpose was to investigate whether different results would be obtained.

Coding schemes

The experimental coders are denoted 1, 2, 3 and 4.

In experiment A the following coding scheme was used:

	Coder constellation			
Material	Production coding	First verification	Second verification	Third verification	Number of schedules
m_1	Coder 1	2	3	4	200
m_2	3	4	1	2	200
m_3	2	3	4	1	200
m_4	4	1	2	3	200

In experiment B each production-coding/first-verification pair from experiment A was used again. Thus, this resulted in

Material	Pair	Remarks
m_1	1,2	no access to code numbers from experiment A; consensus
m_2	3,4	no access to code numbers from experiment A; consensus
m_3	2,3	access to code numbers from experiment A; consensus
m_4	4,1	access to code numbers from experiment A; consensus

In experiment C all four experimental coders were involved. Thus, this resulted in

Material	Coders	Remarks
m_1	1,2,3,4	Access to code numbers from experiment A; consensus
m_2	1,2,3,4	No access to code numbers from experiment A; consensus
m_3	1,2,3,4	Access to code numbers from experiment A; Delphi
m_4	1,2,3,4	No access to code number from experiment A; Delphi

Experiment D consisted of two parts. As to team coding we had the following scheme:

Material	Coders	Number of schedules	Remarks
m_5	1,2,3,4	200	New material; consensus
m_6	1,2,3,4	200	New material; Delphi

As to repeated dependent verification we had the following scheme:

	Coder constellation				
Material	Production coding	First verification	Second verification	Third verification	Number of schedules
m_{51}	1	2	3	4	100
m_{52}	3	4	1	2	100
m_{61}	2	3	4	1	100
m_{62}	4	1	2	3	100

11.2.2 Results

Experiment A gave the results shown in Table 11.11.

Table 11.11 Measuring the quality contribution obtained by repeated dependent verification (percentages within brackets).

Material / Frequencies	m_1		m_2		m_3		m_4	
	Occ	Ind	Occ	Ind	Occ	Ind	Occ	Ind
Number of errors after production coding	48 (14.4)	24 (7.2)	21 (9.5)	20 (9.0)	55 (19.6)	60 (21.4)	48 (12.9)	36 (9.7)
Number of errors after first verification	48 (14.4)	24 (7.2)	21 (9.5)	19 (8.6)	50 (17.9)	48 (17.1)	44 (11.8)	35 (9.4)
Number of errors after second verification	42 (12.6)	20 (6.0)	19 (8.6)	19 (8.6)	47 (16.8)	44 (15.7)	42 (11.3)	34 (9.2)
Number of errors after third verification	43 (12.9)	19 (5.7)	19 (8.6)	20 (9.0)	42 (15.0)	45 (16.0)	39 (10.5)	35 (9.4)
Total number of cases	334	334	222	222	280	280	372	371

Obviously the repeated verification is far from successful. In fact, if we look only at the first verification (which is equivalent to the usual dependent verification) we observe the worst result ever achieved with this kind of control.

The performance seems to differ between coders. For instance, the net effect of coder 2 is very limited: the only effect is small reductions in m_2 and m_4. At the same time, coder 2 has the highest production error rate of all the experimental coders (m_3). Coder 3 corrects most often (but, because of the design, he always corrects the work done by coder 2). On the other hand, coder 3 has the lowest overall production error rate. However, these possible differences between coder are only speculations; it was not the purpose of the study to investigate such things.

Experiment B gave the results shown in Table 11.12.

Table 11.12 Error frequencies with consensus compared to experiment A, access to previous code numbers (%).

Material	m_3		m_4	
	Occ	Ind	Occ	Ind
Consensus	14.4	11.3	13.7	8.5
Experiment A	17.9	17.1	11.8	9.4

The pair working with m_3 did a better job together than separately. The pair working with m_4 did worse with coding of occupation than they did separately (see Table 11.13).

Table 11.13 Error frequencies with consensus compared to experiment A, no access to previous code numbers (%).

Material	m_1		m_2	
	Occ	Ind	Occ	Ind
Consensus	19.9	9.7	9.0	8.1
Experiment A	14.4	7.2	9.5	8.6

One pair performs worse than in experiment A while the other performs slightly better in the consensus situation.

Experiment C gave the results shown in Table 11.14.

Table 11.14 Results of Delphi coding in m_3, access to code numbers.

Result	Number of cases	
	Occupation	Industry
4-0	270	277
3-1	9	3
2-2	1	0
Total	280	280

The corresponding error frequencies for the four coders are given in Tables 11.15 and 11.16.

Table 11.15 Coder error frequencies in m_3, access to code numbers.

Coder	Error frequency (%)	
	Occupation	Industry
1	14.6	11.8
2	16.4	11.8
3	16.1	11.8
4	15.7	11.8

Table 11.16 Coder error frequencies in m_4, no access to code number.

Coder	Error frequency (%)	
	Occupation	Industry
1	10.2	9.2
2	10.5	9.2
3	10.5	8.9
4	10.5	9.7

The experiment showed that group influence is tremendous. Despite anonymity, coding was completely equal in 270 out of 280 cases; i.e., we had a 4-0 result in 96 percent of the occupation coding cases. The associated figure for industry was 277 out of 280, or 99 percent.

These results should be compared with those obtained in experiment A. For occupation the Delphi results are better until the last verification is administered, though the difference is not especially remarkable. However, for industry the Delphi technique produced a clear error reduction. But when the coders did not have access to previous code numbers the results are about equal to those obtained in experiment A.

The consensus technique was carried out for the materials m_1 and m_2. When coding m_1 the coders had access to previous code numbers. The group error rate was 13.9 % for occupation, which is not an improvement compared to the outcome of experiment A. The error rate for industry was 4.2 %, which is better than the associated figures in experiment A. When coding m_2 the group had no access to previous code numbers. No remarkable differences could be discovered.

Experiment D gave the results shown in Table 11.17.

Table 11.17 Measuring the quality contribution by repeated dependent verification. Error frequency (%).

Material / Frequencies %	m_5 Occ	Ind
Errors after production coding	12.4	9.9
Errors after first verification	12.0	10.3
Errors after second verification	11.1	7.9
Errors after third verification	9.5	8.3

These results can be compared with those obtained with consensus: for the same material we have group error frequencies 13.6 % for occupation and 8.7 % for industry.

The Delphi technique gave better results than repeated verification. Consider Tables 11.18 and 11.19.

Table 11.18 Error frequencies (%) using Delphi, no access to previous code numbers.

Material / Coder	m_6 Occ	Ind
1	11.2	7.6
2	10.8	8.0
3	11.2	7.6
4	11.7	8.5

Table 11.19 Measuring the quality contribution by repeated dependent verification. Error frequency (%).

Material / Frequencies %	m_6 Occ	Ind
Errors after production coding	14.6	10.4
Errors after first verification	13.5	9.5
Errors after second verification	14.9	9.9
Errors after third verification	13.5	9.9

11.2.3 Summary

The dependent verification produced unusually meagre results. The coders had a remarkable tendency to accept code numbers already assigned. The 50 percent rule of thumb did not work at all: at most we got 20 percent error reduction and at the other extreme we even obtained negative results between verifications. This is exceptional.

Neither consensus nor Delphi coding gave any clearly advantageous results. Thus we have no indication that we have found a new approach in this area, and as a consequence we have not analyzed the experimental results in more detail to look for significances and so forth. However, if in a special case the cost situation should be very favourable one might consider consensus or Delphi an option.

A detailed description of the experiments is found in Lyberg and Tillquist (1974).

12 Independent verification

In Dalenius and Frank (1968) and in Lyberg (1969) several alternative schemes for independent verification are discussed. The schemes differ in complexity and in cost and probability characteristics. In this chapter some of these schemes are presented in detail, and then independent verification is evaluated in general.

12.1 Some basic concepts

a) We consider a population of N elements $(E_1, E_2, \ldots, E_N)$.

b) Each of these elements belongs to one and only one of K categories. Furthermore, N_i is the total number of elements belonging to category i, where i = 1, 2, ..., K. Thus

$$\sum_{i=1}^{K} N_i = N.$$

The corresponding relative frequencies are denoted $r_i = N_i/N$.

c) We have at our disposal sets of coders denoted

$$C = \{C_1, C_2, \ldots, C_M\}$$

$$B = \{B_1, B_2, \ldots, B_L\}$$

We have also a referee A.

In practice it is assumed that A performs better than the B-coders and that the B-coders perform better than the C-coders, but this assumption is not necessary for the subsequent calculations.

d) For a given element E_G in the population we define the following code numbers:

$a(E_G) = a$ is the code number assigned by the referee A,

$b_J(E_G) = b_J$ is the code number assigned by coder B_J selected at random from the set B,

$c_J(E_G) = c_J$ is the code number assigned by coder C_J selected at random from the set C,

$z(E_G) = z$ is the outgoing code number; i.e. the resulting code number of the system, and

$\theta(E_G) = \theta$ is the true code number.

The following notations for probabilities and costs are introduced:

	Set of coders		
	A	B	C
$P\{\text{assigned code number} = j \mid \theta=i\}$	t_{ij}	q_{ij}	p_{ij}
$P\{\text{assigned code number} = i \mid \theta=i\}$	t_{ii}	q_{ii}	p_{ii}
Average cost for assigning one code number	R_A	R_B	R_C

For example, p_{ij} is the probability that a C-coder assigns the element to category j when in fact the element belongs to category i. Thus, it is assumed that

$P\{b_J = j \mid \theta=i\} = q_{ij}$ is the same for all B-coders and that
$P\{c_J = j \mid \theta=i\} = p_{ij}$ is the same for all C-coders.

Another assumption adopted in the actual construction of the scheme is that $R_A > R_B > R_C$, but like the coder performance assumption mentioned above, this is not necessary for the subsequent calculations.

From our pool of coders {A, B, C} we will construct different formations of coding teams. The team formation together with the coding decision rule define the coding scheme. Seven such schemes are presented. In all schemes it is assumed that all code numbers are assigned independently by the coders involved.

The schemes presented seem to have a realistic design. As a matter of fact, scheme number one is very similar to the US Bureau of the Census' three-way independent verification system used in the 1960 US Censuses of Population and Housing.

We have assumed that there is no dependence between the code numbers assigned by different coders. Although coders with the same training, for instance, or coders working under the same crew leader certainly have a tendency to make the same errors such dependences have not been taken into account for in this presentation. On the contrary, the schemes are based upon the assumption that a code number resulting from the decision rule (majority or modal) is highly unlikely to be incorrect.

12.2 Seven independent verification schemes

Each scheme is described by a definition of the team formation and the coding decision rule. The details of the associated functions of probability and cost are presented in appendix I immediately following this chapter.

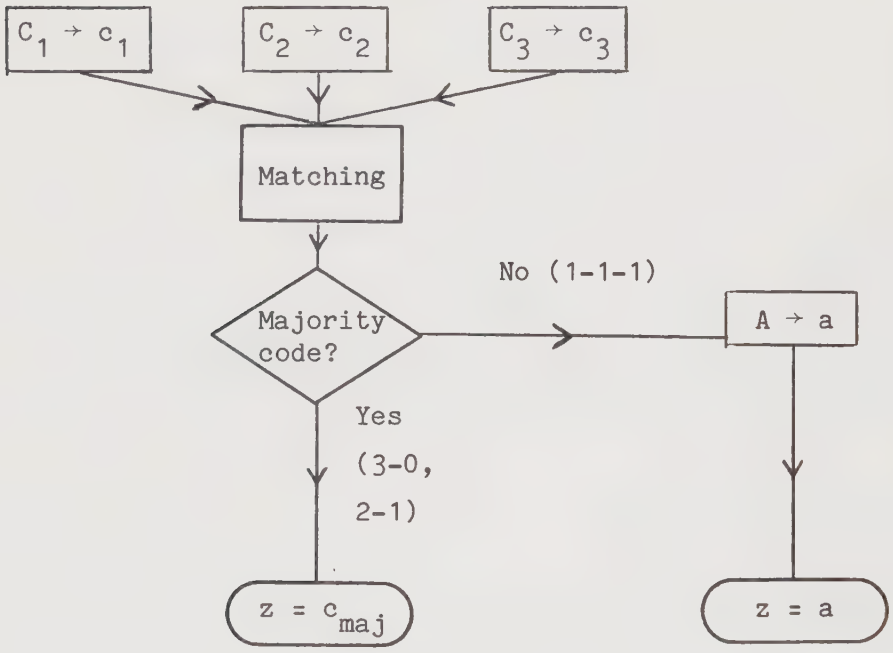
Scheme no. 1 (S1)

Team formation

The coding team consists of three coders chosen at random from the set C and the referee A. The code numbers assigned by these coders are denoted c_1, c_2, c_3, and a respectively.

The coding decision rule

The rule is diagrammed below:



This scheme could be called three-way independent verification.

Thus

If $c_1 = c_2 = c_3 = j$ then j is the outgoing code number.
This is the 3-0 case.

If $c_1 \neq c_2 = c_3 = j$, or
$c_2 \neq c_1 = c_3 = j$, or
$c_3 \neq c_1 = c_2 = j$, then j is the outgoing code number.
These are the 2-1 cases.

If $c_1 \neq c_2 \neq c_3 \neq c_1$ then a is the outgoing code number.
This is the 1-1-1 case.

It can be shown that the probability

P {the outgoing code number is true} =

$$= \sum_i r_i t_{ii} - \sum_i r_i \left[t_{ii} \sum_j p_{ij}^2 (3-2\, p_{ij}) - p_{ii}^2 (3-2\, p_{ii}) \right]$$

and that the expected cost of the coding operation is

$$R_{S1} = N(3R_C + (1 - \sum_i r_i \sum_j p_{ij}^2 (3-2\, p_{ij}))\, R_A)$$

Of course, scheme S1 could be carried out by a team of B-coders and an A-coder. Then p_{ij} and R_C should be exchanged with q_{ij} and R_B respectively.

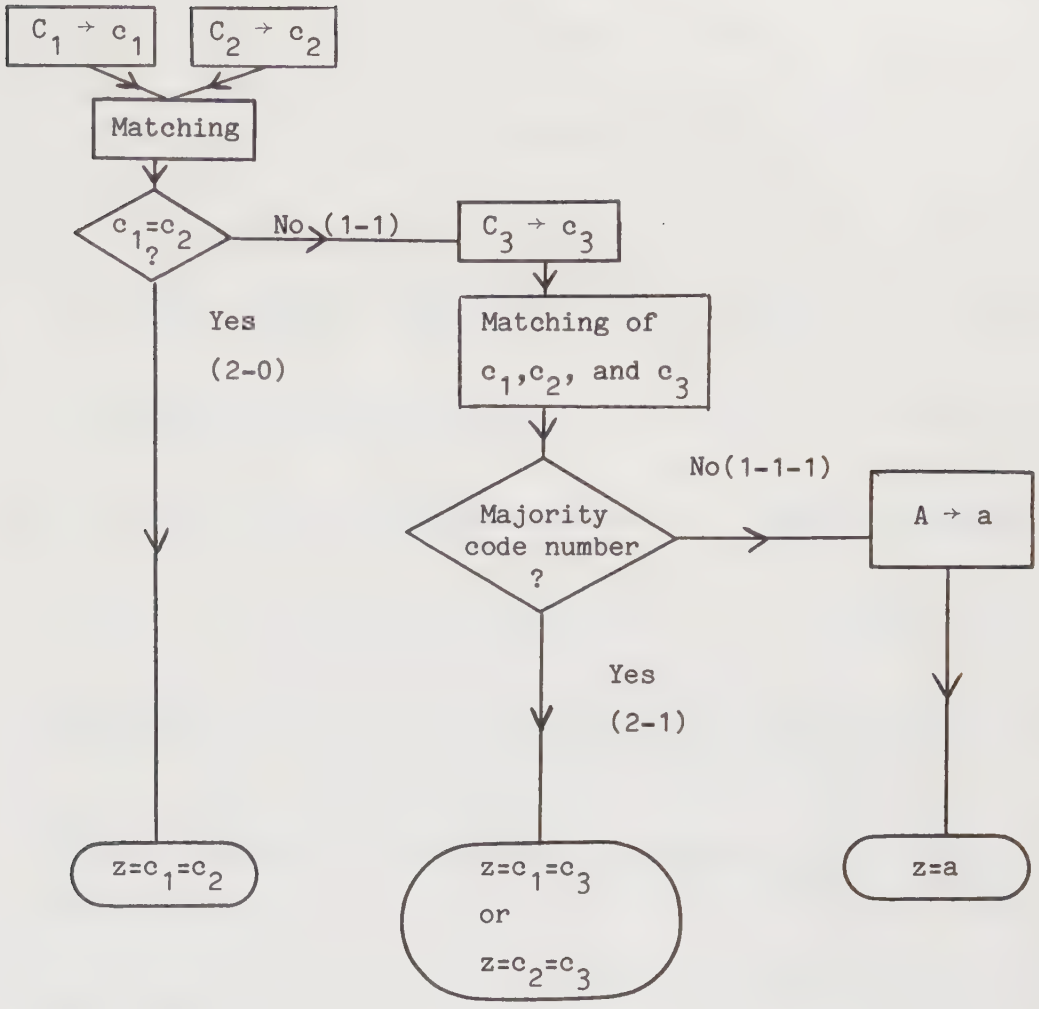
Scheme no. 2 (S2)

Team formation

The coding team is the same as in S1.

The coding decision rule

The rule is diagrammed below:



Thus

If $c_1=c_2=j$ then j is the outgoing code number. This is the 2-0 case.

If $c_1 \neq c_2$ then the element is referred to C_3.
If $c_1 \neq c_2=c_3=j$, or
$c_2 \neq c_1=c_3=j$, then j is the outgoing code number. These are the 2-1 cases.

If $c_1 \neq c_2 \neq c_3 \neq c_1$ then a is the outgoing code number. This is the 1-1-1 case.

This is a sequential scheme starting with only two coders and ending up with the majority rule in S1.

It can be shown that the probability

P {the outgoing code number is true} =

$$= \sum_i r_i t_{ii} - \sum_i r_i \left[t_{ii} \sum_j p_{ij}^2 \, (3-2\, p_{ij}) - p_{ii}^2 \, (3-2\, p_{ii}) \right]$$

and that the expected cost for the coding operation is

$$R_{S2} = N \left\{ \left(3 - \sum_i r_i \sum_j p_{ij}^2\right) R_C + \left(1 - \sum_i r_i \sum_j p_{ij}^2 \, (3-2\, p_{ij})\right) R_A \right\}$$

Scheme no. 3 (S3)

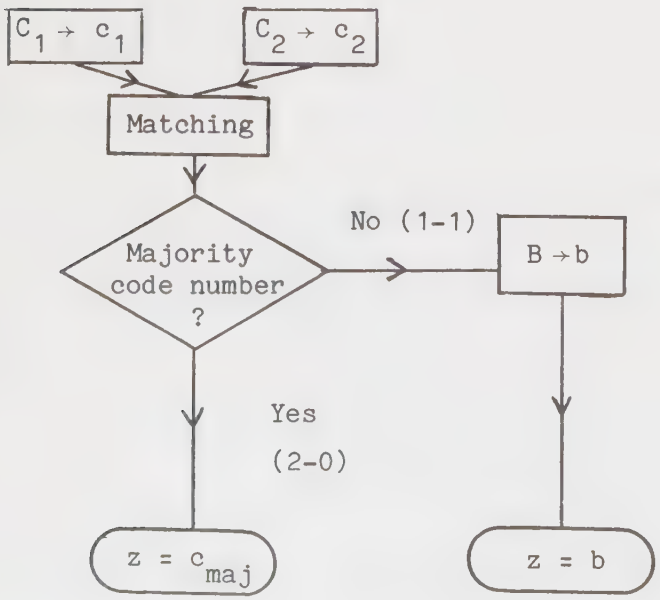
Team formation

The coding team consists of one B-coder and two C-coders chosen at random from the sets B and C respectively.

The code numbers assigned by these coders are denoted b, c_1, and c_2 respectively.

The coding decision rule

The rule is diagrammed below:



Thus

If $c_1 = c_2 = j$ then j is the outgoing code number. This is the 2-0 case.

If $c_1 \neq c_2$ then the element is referred to B and $b = j$ is the outgoing code number. This is the 1-1 case.

It can be shown that the probability

P {the outgoing code number is true} =

$$= \sum_i r_i q_{ii} - \sum_i r_i \{q_{ii} \sum_j p_{ij}^2 - p_{ii}^2\}$$

and that the expected cost for the coding operation is

$$R_{S3} = N \{2 R_C + (1 - \sum_i r_i \sum_j p_{ij}^2) R_B\}$$

Scheme no. 4 (S4)

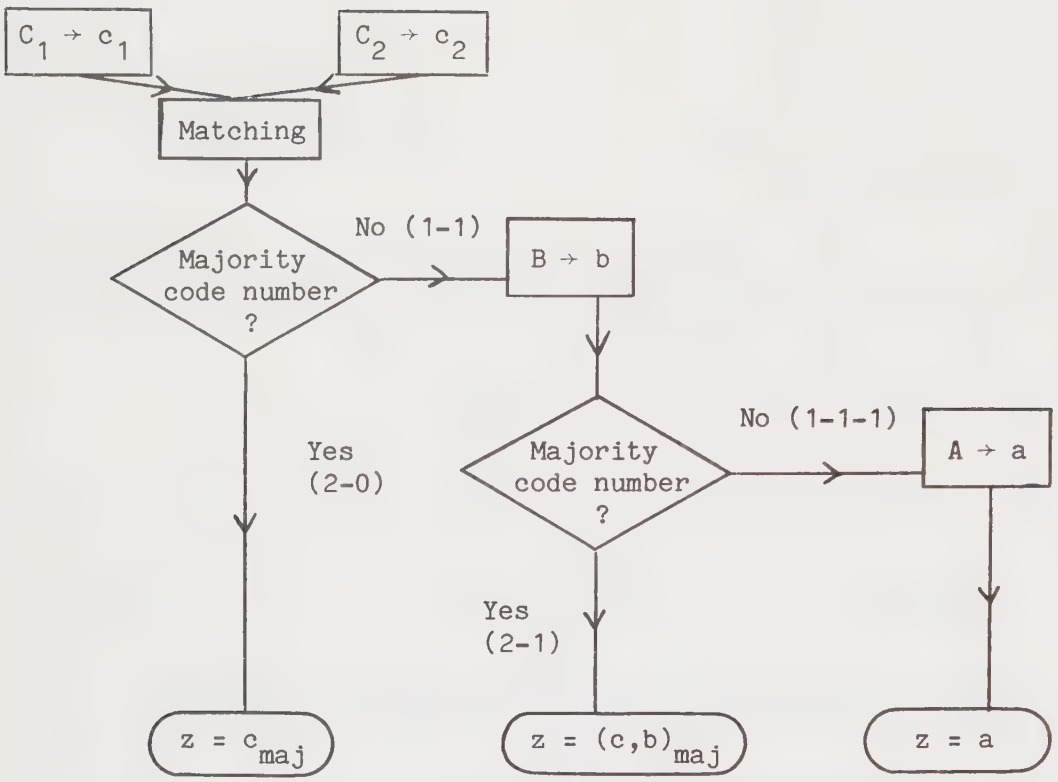
Team formation

The coding team consists of one B-coder, two C-coders, and the A-coder. The B- and C- coders are chosen randomly from the subsets B and C.

The code numbers assigned by these coders are denoted b, c_1, c_2, and a respectively.

The coding decision rule

The rule is diagrammed below:



Thus

If $c_1 = c_2 = j$ then j is the outgoing code number. This is the 2-0 case.

If $c_1 \neq c_2$ then the element is referred to B and b is compared with c_1 and c_2. If $b = c_1 = j$ or if $b = c_2 = j$ then $b = j$ is the outgoing code number. These are the 2-1 cases.

If $b \neq c_1 \neq c_2 \neq b$ the element is referred to A and a is the outgoing code number. This is the 1-1-1 case.

It can be shown that the probability

P {the outgoing code number is true} =

$$= \sum_i r_i t_{ii} - \sum_i r_i \{t_{ii} \sum_j p_{ij}^2 - p_{ii}^2 + 2\, t_{ii} \sum_j q_{ij} p_{ij} (1-p_{ij})$$

$$- 2\, q_{ii} p_{ii} (1-p_{ii})\}$$

and that the expected cost for the coding operation is

$$R_{S4} = N [2 R_C + (1-\sum_i r_i \sum_j p_{ij}^2) R_B$$

$$+(1-\sum_i r_i \sum_j p_{ij} (p_{ij} + 2 q_{ij} (1-p_{ij})))R_A]$$

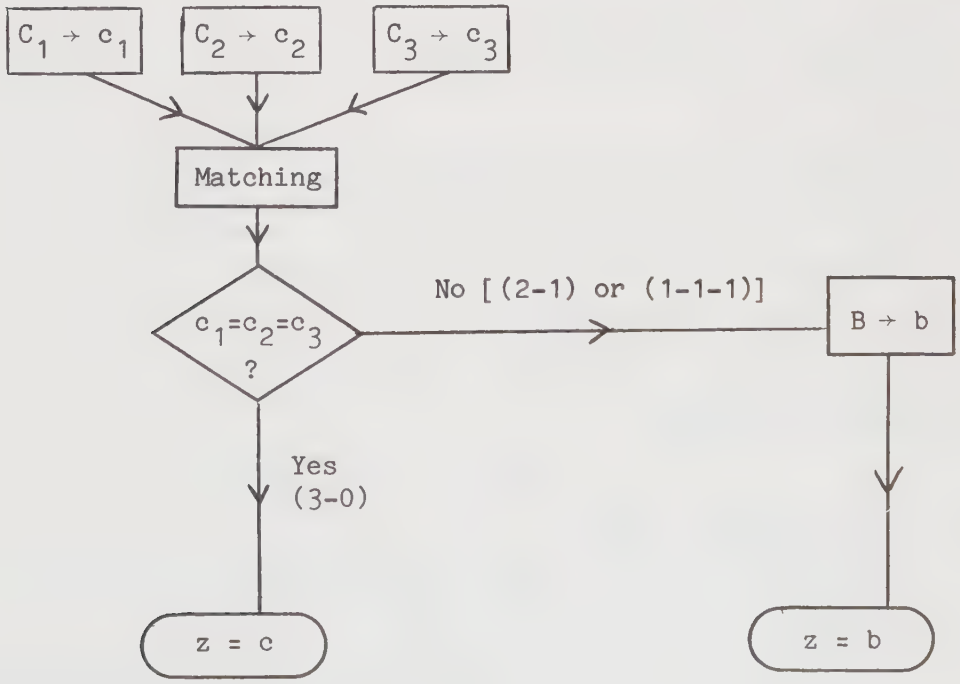
Scheme no. 5 (S5)

Team formation

The coding team consists of one B-coder and three C-coders. The B and C-coders are chosen randomly from the subsets B and C. The code numbers assigned by these coders are denoted b, c_1, c_2, and c_3 respectively.

The coding decision rule

The rule is diagrammed below:



Thus

If $c_1 = c_2 = c_3 = j$ then j is the outgoing code number. This is the 3-0 case.

Otherwise the element is referred to the B-coder and b = j is the outgoing code number. These are the 2-1 or 1-1-1 cases.

It can be shown that the probability

P {the outgoing code number is true} =

$$= \sum_i r_i q_{ii} - \sum_i r_i \left[q_{ii} \sum_j p_{ij}^3 - p_{ii}^3 \right]$$

and that the expected cost for the coding operation is

$$R_{S5} = N \left(3 R_C + \left(1 - \sum_i r_i \sum_j p_{ij}^3 \right) R_B \right)$$

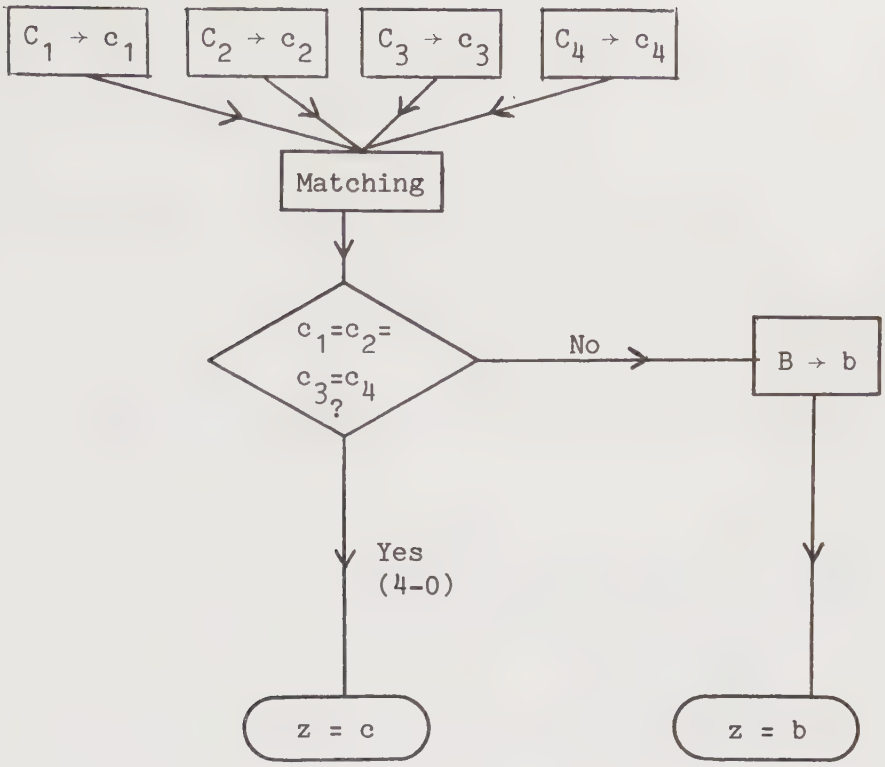
Scheme no. 6 (S6)

Team formation

The coding team consists of one B-coder and four C-coders. The B- and C-coders are chosen randomly from the subsets B and C. The code numbers assigned by these coders are denoted b, c_1, c_2, c_3, and c_4 respectively.

The coding decision rule

The rule is diagrammed below:



Thus

If $c_1 = c_2 = c_3 = c_4 = j$ then j is the outgoing code number.

Otherwise the element is referred to the B-coder and b = j is the outgoing code number.

It can be shown that the probability

P {the outgoing code number is true} =

$$= \sum_i r_i q_{ii} - \sum_i r_i [q_{ii} \sum_j p_{ij}^4 - p_{ii}^4]$$

and that the expected cost for the coding operation is

$$R_{S6} = N \left(4 R_C + (1 - \sum_i r_i \sum_j p_{ij}^4) R_B\right)$$

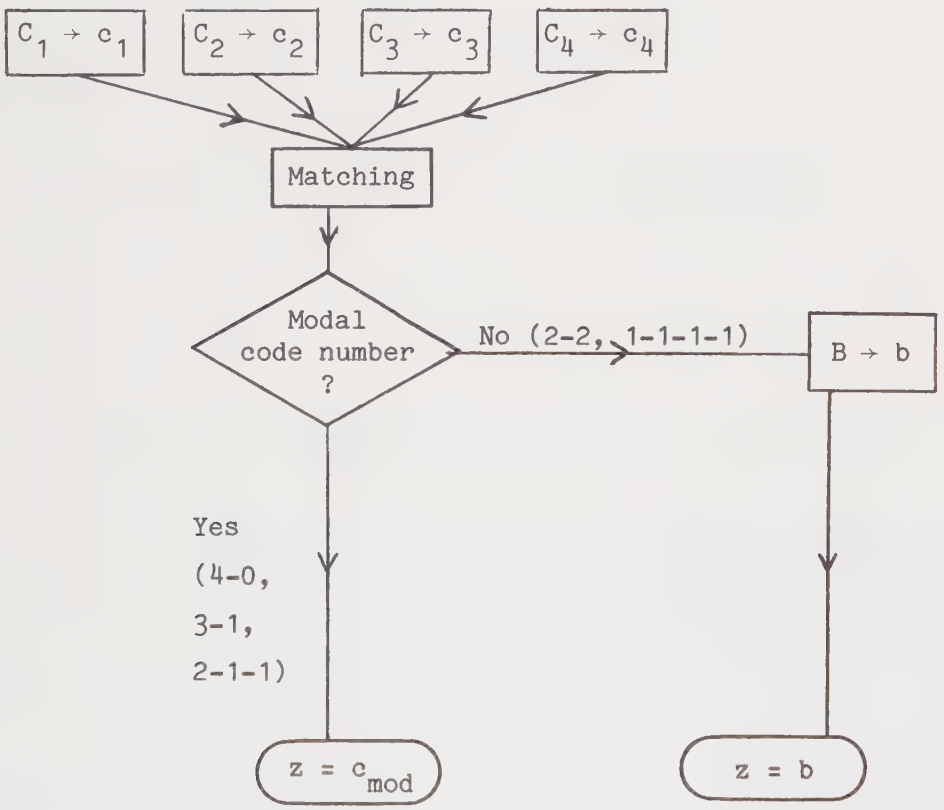
Scheme no. 7 (S7)

Team formation

The formation is the same as in scheme no. 6.

The coding decision rule

The rule is diagrammed below. Here we are dealing with the modal rule.



Thus

If $c_1 = c_2 = c_3 = c_4 = j$ then j is the outgoing code number. This is the 4-0 case.

If $c_1 \neq c_2 = c_3 = c_4 = j$,
$c_2 \neq c_1 = c_3 = c_4 = j$,
$c_3 \neq c_1 = c_2 = c_4 = j$

or

$c_4 \neq c_1 = c_2 = c_3 = j$ then j is the outgoing code number. These are the 3-1 cases.

If $c_1 \neq c_2 \neq c_3 \neq c_1$ but $c_3 = c_4 = j$,
$c_1 \neq c_3 \neq c_2 \neq c_1$ but $c_2 = c_4 = j$,
$c_1 \neq c_4 \neq c_3 \neq c_1$ but $c_3 = c_2 = j$,
$c_3 \neq c_4 \neq c_1 \neq c_3$ but $c_1 = c_2 = j$,
$c_4 \neq c_2 \neq c_3 \neq c_4$ but $c_3 = c_1 = j$,
or
$c_2 \neq c_3 \neq c_1 \neq c_2$ but $c_1 = c_4 = j$,

then j is the outgoing code number. These are the 2-1-1 cases.

If $c_1 = c_2 = l$ and $c_3 = c_4 = m$, $l \neq m$,
$c_1 = c_3 = l$ and $c_2 = c_4 = m$, $l \neq m$,
or
$c_1 = c_4 = l$ and $c_3 = c_2 = m$, $l \neq m$,

then the element is referred to the B-coder and b = j is the outgoing code number. These are the 2-2 cases.

If all c differ, then the element is referred to the B-coder and b = j is the outgoing code number. This is the 1-1-1-1 case.

It can be shown that the probability

P {the outgoing code number is true} =

$$= \sum_i r_i p_{ii}^4 + 4 \sum_i r_i p_{ii}^3 (1-p_{ii})$$

$$+ 6 \sum_i r_i p_{ii}^2 (1-2 p_{ii} + 2 p_{ii}^2 - \sum_j p_{ij}^2)$$

$$+ \sum_i r_i q_{ii} [1-9 \sum_j p_{ij}^4 + 8 \sum_j p_{ij}^3 - 6 \sum_j p_{ij}^2 + 6(\sum_j p_{ij}^2)^2]$$

and that the expected cost for the coding operation is

$$R_{S7} = N[4R_C + \sum_i r_i[6(\sum_j p_{ij}^2)^2 - 9\sum_j p_{ij}^4 - 6\sum_j p_{ij}^2 + 8\sum_j p_{ij}^3 + 1]]R_B$$

12.3 A comparison of the different schemes

The different schemes can be compared with respect to $P(z=\theta)$ and $R_{S.}$; i.e. the probability of the true outgoing code number and the cost of obtaining it. The separate cost factors R_A, R_B and R_C are considered constant in all schemes and as a consequence the cost comparisons suffer from some lack of realism. When a scheme is applied one has to remember that, even if it is the best in terms of $P(z=\theta)$ and R_S, the administrative costs also enter in. For instance, when comparing a three-way non-sequential system with a sequential one we can be sure that the administrative costs (extra matching, record-keeping, computer listings) make the costs of the systems more nearly equal than is indicated by our comparisons. It would be desirable to add a special administrative cost term to each R_S. This has not been done, mainly because such terms are difficult to estimate and vary considerably depending on available computer facilities, how the flow of schedules is arranged, and so forth. Experience tells us that a 100 % manual system generates heavy administrative costs. The costs are cut when the computer is used, and when coding is carried out on computer terminals with a screen display the administrative costs can be cut even more. So let us assume that we have such a terminal coding system in comparing the schemes presented above.

We have constructed a computer program for comparisons between the seven different schemes. In this special illustration the following four sets of values are fixed:

K = 2	K = 5	K = 10	K = 50
N = 10 000	N = 10 000	N = 10 000	N = 10 000
r_i = .5	r_i = .2	r_i = .1	r_i = .02

Three different sets of values for R_A, R_B and R_C are considered:

Set I: $R_A = R_B = R_C = .5$

Set II: $R_A = .6,\ R_B = .4,\ R_C = .3$

Set III $R_A = .9,\ R_B = .6,\ R_C = .3$

Within each set the probabilities take the following values.

p_{ii} .80; .82; ...; 1.00

q_{ii} .90; .92; ...; 1.00

t_{ii} .96; .98; 1.00

It is assumed that all the erroneous code numbers have the same probability of occurrence, i.e.:

$$p_{\substack{ij \\ i\neq j}} = \frac{1 - p_{ii}}{K-1}$$

$$q_{\substack{ij \\ i\neq j}} = \frac{1 - q_{ii}}{K-1} \qquad \text{and}$$

$$t_{\substack{ij \\ i\neq j}} = \frac{1 - t_{ii}}{K-1}$$

In the subsequent discussion (12.3.1-12.3.3) we restrict ourselves to the case in which K = 10. As can be seen in the tables in appendix II, though, the conclusions depend on the value of K. Probably, the conclusions will also depend on the choice of assumptions as to r_i, p_{ij}, q_{ij} and t_{ij}. For instance, in a practical situation it is highly unlikely that all p_{ij} $(i\neq j)$ are equal. Thus the following is merely an illustration.

12.3.1 Comparing S1 and S2

Let us first examine S1 and S2, the only schemes using a combination of the referee A and the C-coders. We have already seen that those two schemes give identical probabilities for the true outgoing code number. Thus the remaining factor is the cost and since S1 always utilizes the third C-coder, S2 is a less expensive and equally good alternative to the scheme used in the 1960 U S Censuses of Population and Housing.

Consider Table 12.1.

Table 12.1 Probabilities and costs for schemes S1 and S2.

P_{ii}	P{true outgoing code number}			R_{S1}			R_{S2}		
	$t_{ii}=1$	$t_{ii}=.98$	$t_{ii}=.96$	Cost set I	II	III	Cost set I	II	III
.80	.9869	.9851	.9833	15 454	9 545	9 818	12 232	7 612	7 884
.82	.9893	.9878	.9863	15 374	9 449	9 674	11 994	7 421	7 646
.84	.9916	.9904	.9892	15 301	9 361	9 542	11 759	7 236	7 416
.86	.9935	.9926	.9916	15 234	9 281	9 422	11 525	7 056	7 196
.88	.9952	.9946	.9939	15 175	9 210	9 315	11 295	6 882	6 987
.90	.9967	.9962	.9957	15 123	9 148	9 222	11 068	6 715	6 789
.92	.9979	.9976	.9973	15 080	9 096	9 144	10 845	6 555	6 603
.94	.9988	.9986	.9984	15 046	9 055	9 083	10 626	6 403	6 431

This table shows that the cost difference between S1 and S2 is substantial and increases when p_{ii} approaches 1. The difference between the costs is the same in the cost sets II and III because $R_{S1}-R_{S2}$ depends only on R_C, and $R_C=$ .3 in both II and III.

12.3.2 Comparing S3 - S7

The probabilities p_{ii} and q_{ii} are present in S3 - S7. S4 also includes t_{ii}. Let us investigate what happens when p_{ii} is varied together with q_{ii} (and with t_{ii} in S4). Probabilities and costs for schemes nos. 3, 4, 5, 6 and 7 are calculated in the example and tables are presented in appendix II of this chapter. Please note that p, q, and t have some values that contradict our assumption that A always performs better than B and that B always performs better than C. (The tables in appendix II give an indication of what will happen when this assumption is not fulfilled.)

The tables in appendix II show that for K = 10, S5 and S6 cannot compete with the rest: they are more expensive and give lower quality. To let three or more C-coders code simultaneously is apparently a waste of resources. Sequential methods are better. Then schemes nos. 3, 4, and 7 remain. Of these, S7 is the most expensive. It is better than S3 in probability considerations and about equal when compared with S4, but the cost difference is large. Thus we are left with schemes nos. 3 and 4. The choice between them must be based on the desired outgoing quality for the coding process. S3 is the cheaper of the two but has a lower probability of correct outgoing code number. The cost of S3 is independent of the q_{ii}-values, while the cost of S4 decreases somewhat when q_{ii} increases.

12.3.3 Comparing all the schemes

The different schemes involve the following coder combinations.

Scheme	Coder combination
1	A, C
2	A, C
3	B, C
4	A, B, C
5	B, C
6	B, C
7	B, C

We present in Tables 12.2 and 12.3 a few of the values in the tables presented in appendix II. First a probability comparison is made and then the corresponding cost comparison is carried out.

Table 12.2 Probability comparisons when $q_{ii} = .90$.

p_{ii}	S1 and S2			S3	S4			S5	S6	S7
	$t_{ii}=1$	$t_{ii}= .98$	$t_{ii}= .96$		$t_{ii}=1$	$t_{ii}= .98$	$t_{ii}= .96$			
.80	.9869	.9850	.9832	.9600	.9912	.9899	.9887	.9511	.9410	.9877
.82	.9893	.9978	.9863	.9640	.9925	.9914	.9903	.9551	.9452	.9907
.84	.9916	.9904	.9892	.9680	.9937	.9927	.9918	.9592	.9498	.9932
.86	.9935	.9926	.9917	.9720	.9948	.9940	.9932	.9636	.9547	.9953
.88	.9952	.9945	.9938	.9760	.9958	.9951	.9945	.9681	.9600	.9968
.90	.9967	.9962	.9957	.9800	.9967	.9962	.9957	.9729	.9656	.9980
.92	.9979	.9976	.9972	.9840	.9975	.9972	.9968	.9779	.9716	.9989
.94	.9988	.9986	.9984	.9880	.9983	.9980	.9977	.9831	.9781	.9995
.96	.9995	.9994	.9993	.9920	.9989	.9988	.9986	.9885	.9849	.9998
.98	.9999	.9998	.9998	.9960	.9995	.9994	.9994	.9941	.9922	1.0000
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Table 12.3 Probability comparisons when $q_{ii} = .96$.

p_{ii}	S1 and S2			S3	S4			S5	S6	S7
	$t_{ii}=1$	$t_{ii}= .98$	$t_{ii}= .96$		$t_{ii}=1$	$t_{ii}= .98$	$t_{ii}= .96$			
.80	.9869	.9850	.9832	.9813	.9938	.9929	.9920	.9804	.9764	.9899
.82	.9893	.9878	.9863	.9834	.9948	.9941	.9933	.9820	.9781	.9924
.84	.9916	.9904	.9892	.9855	.9958	.9951	.9945	.9837	.9799	.9945
.86	.9935	.9926	.9917	.9875	.9966	.9961	.9956	.9854	.9819	.9962
.88	.9952	.9945	.9938	.9894	.9973	.9969	.9965	.9872	.9840	.9975
.90	.9967	.9962	.9957	.9913	.9980	.9977	.9974	.9891	.9862	.9985
.92	.9979	.9976	.9972	.9932	.9986	.9984	.9981	.9911	.9887	.9992
.94	.9988	.9986	.9984	.9950	.9991	.9989	.9988	.9932	.9912	.9996
.96	.9995	.9994	.9993	.9967	.9995	.9994	.9993	.9954	.9940	.9999
.98	.9999	.9998	.9998	.9984	.9998	.9997	.9997	.9976	.9969	1.0000
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

The costs of the different schemes (see Table 12.4) depend solely on the p_{ii}-value within each cost set (with the exception of S4 where the q_{ii}-value also has a small impact on the cost; however, the effect is so small that it is ignored in the summar table below).

Table 12.4 Cost comparison.

Cost set	p_{ii}	Scheme 1	2	3	4	5	6	7
I	.80	15 454	12 232	11 778	≈ 12 000	17 440	22 952	20 178
	1.00	15 000	10 000	10 000	10 000	15 000	20 000	20 000
II	.80	9 545	7 612	7 422	≈ 7 700	10 952	14 362	12 142
	1.00	9 000	6 000	6 000	6 000	9 000	12 000	12 000
III	.80	9 818	7 884	8 133	≈ 8 600	11 927	15 542	12 213
	1.00	9 000	6 000	6 000	6 000	9 000	12 000	12 000

All of these cost functions could be represented by straight lines (although they are not exactly straight). We have already noticed that schemes 3 and 4 are less costly than the others in the group S3 - S7. S3 is cheaper than S4 but S4 produces better coding quality. How do these schemes compete with S2? The S2-probabilities are found to lie between those of S3 and S4, and the cost for S2 competes very well with those of S3 and S4. Here the cost set plays an important role.

It seems that S2, S3, and S4 is the most realistic set of scheme They are the best in terms of cost and quality, and they are realistic in the sense that they could be used in a production coding.

The choice between these three schemes depends on the desired outgoing quality of the coding operation. The stochastic charac-teristics and the cost functions depend on the parameter values. The probability functions cross each other at certain points.

Thus the probabilities must be estimated if we want an efficient team formation. Such estimates could be computed during the qualification stage and during the production stage, for instance, by means of a master set.

12.4 Special considerations

Independent verification is a more reliable process than dependent verification: it generally allows more accurate estimation of errors and the coding results are more credible. However, the advantages of dependent verification, listed in Linebarger et al. (1976), simultaneously constitute the drawbacks of independent verification. Thus the latter is more expensive, more disruptive to handle, and more time-consuming. Of course the cost factor is very important (that was the reason why independent verification was not used throughout the entire 1970 Swedish Census of Population). There is also a certain delay in feedback operations with independent verification. For instance, there is a tendency not to discover deteriorating coder performance until rather late, and correction measures are thus delayed. On the other hand, more errors are detected and the picture of the error structure is clearer with independent verification.

The basic idea in all independent schemes is that the outgoing code number (majority or modal) is very likely to be the correct one. Thus very seldom will a minority coder be unjustly charged with an error. However, it is important that the coders whose code numbers are compared be of approximately equal ability. In Harris (1974) it is pointed out that two "poor" coders could overrule a "good" coder simply because the former have not read the instructions properly. Harris gives an example from Mortality Medical Coding which is operated under a sequence of coding instructions. The example goes like this.

> If diagnosis X is listed, code 111. A "poor" coder might stop here, assign 111 and go on to the next coding unit. However, there may be an additional instruction which says whenever code number 111 appears, check to see if diagnosis Y is also listed. If yes, change the code number to 123. The "good" coder often arrives at 123 and as a "reward" he is charged with an error.

The hypothesis that "poor" coders adversely affect "good" coders has been tested by the staff at the US Department of Health, Education, and Welfare. The hypothesis was rejected. However, it is obvious that this case must occur now and then; i.e. the probability of making the same error could be substantial. For instance, during a Census of Population additional coding instructions are produced continuously (at least in Sweden) or at least as soon as "new cases" occur. Obviously there is a risk that less ambitious coders might fail to learn the new instructions quickly and might as a result overrule a more dutiful coder.

In Boston (1977 a) some doubts are thrown on the basic idea of independent verification. An expert evaluation of cases of coding agreement shows that the error rate in occupation coding cannot be neglected: a series of tests showed that the majority code number was incorrect in .7 to 1.8 per cent of the cases. However, everything depends on what is meant by "highly likely to be correct". Compared to the error rates obtained with dependent verification we might still be in a favorable position with independent verification. Besides, in this series of tests only one expert was used to determine if the agreement cases are correct or not. Experience shows that there are no experts in complex coding (by "experts" we mean persons with unit probability of correct coding). This lack of experts create both theoretical and practical problems.

The use of independent verification schemes often requires great administrative resources. In order to secure the element of independence the code numbers must be assigned in a proper manner. There are several options for doing this, depending on the scale of the operation and the financial resources avail able. The independent code numbers could be assigned in at least the following ways:

a) by masking: the production code number to be verified is masked and the independent code number could be assigned in connection with the description;

b) by transcription: the description is copied and the indepen- dent code number is assigned in connection with the descrip- tion;

c) by separate forms: independent code numbers are assigned on separate forms with an id corresponding to the schedule containing the description. (This could be in optical character recognition form, a technique which certainly makes things easier. Matching and production quality control statistics can be taken care of by the computer.)

d) by computer terminal coding: here the coder keys in his code numbers at a computer terminal and the computer matches, carries out logical checks, and makes things easier all the way.

Nowadays only c) and d) are used at Statistics Sweden.

Despite what has been said about error rates in agreement cases independent verification is the best system we have today in terms of quality. However, the administrative apparatus require and the costs can be large. For instance, the US Bureau of the Census was in big trouble in the 1970 Census due to problem with the mechanical matching and computer processing. As a

result the independent system had to be abandoned in the place of work and the general coding and a manual independent verification system was utilized in industry and occupation coding in that Census. The Swedish experiences have been better: in the 1970 Census an optical character system was used and in the 1975 Census the coding was carried out on computer terminals. In 1970 we had to abandon our plans for an entirely independent operation for financial reasons, but the 1975 operation went smoothly although the computer system went down now and then.

Experience shows that among the independent schemes investigated the simple ones are the best both from a practical and a theoretical point of view. However, much remains to be done in the realm of scheme construction. The probabilistic assumtions are very simple in the schemes presented above. More realistic (but thus more complex) schemes are desirable.

APPENDIX I

PROBABILITIES AND COSTS FOR
INDEPENDENT VERIFICATION SCHEMES

S1

Probabilities

The probabilities for the different cases are

$$P\{3\text{-}0|\theta=i\} = p_{ij}^3$$

$$P\{2\text{-}1|\theta=i\} = 3\, p_{ij}^2(1-p_{ij})$$

$$\begin{aligned}
P\{1\text{-}1\text{-}1|\theta=i\} &= \Sigma_j p_{ij} \sum_{l\neq j} p_{il} \sum_{v\neq l,j} p_{iv} \\
&= \Sigma_j p_{ij} \sum_{l\neq j} p_{il}(1-p_{il}-p_{ij}) \\
&= \Sigma_j p_{ij}\{\Sigma_l p_{il}(1-p_{il}-p_{ij}) - p_{ij}(1-2\,p_{ij})\} \\
&= \Sigma_j p_{ij}\{\Sigma_l (p_{il} - p_{il}^2 - p_{il}p_{ij}) - p_{ij} + 2\,p_{ij}^2\} \\
&= \Sigma_j p_{ij}\{1-\Sigma_l p_{il}^2 - p_{ij}\sum_l p_{il} - p_{ij} + 2\,p_{ij}^2\} \\
&= \Sigma_j p_{ij}\{1-\Sigma_j p_{ij}^2 - 2p_{ij} + 2p_{ij}^2\} \\
&= 1-3\Sigma_j p_{ij}^2 + 2\Sigma_j p_{ij}^3 = 1-\Sigma_j p_{ij}^2(3-2p_{ij})
\end{aligned}$$

Thus

$$\begin{aligned}
P\{z=j|\theta=i\} &= P\{(3\text{-}0)\to j|\theta=i\} + P\{(2\text{-}1)\to j|\theta=i\} + \\
&\quad + P\{a=j|\theta=i\}P(1\text{-}1\text{-}1|\theta=i) \\
&= p_{ij}^3 + 3p_{ij}^2(1-p_{ij}) + t_{ij}(1-\Sigma_j p_{ij}^2(3-2p_{ij}))
\end{aligned}$$

and we get

P{the outgoing code number is true} =

$$= \sum_{i=1}^{K} r_i \, P(z=i|\theta=i)$$

$$= \sum_i r_i\{3p_{ii}^2 - 2p_{ii}^3 + t_{ii}(1-\sum_j p_{ij}^2(3-2p_{ij}))\}$$

$$= \sum_i r_i t_{ii} - \sum_i r_i\{t_{ii} \sum_j p_{ij}^2(3-2p_{ij}) - p_{ii}^2 (3-2p_{ii})\}$$

Sometimes t_{ii} is unrealistically assumed to be 1. Then

$$P(z=\theta|t_{ii}=1) = 1 - \sum_i r_i \sum_{j\neq i} p_{ij}^2(3-2p_{ij})$$

Costs

The coding is carried out at two stations. At the first station all elements are coded by the C-coders. Thus 3N elements are coded there. At the second station N'_{S1} elements are coded by the A-coder and the expected number of N'_{S1} is

$$E(N'_{S1}) = N(1 - \sum_i r_i \sum_j p_{ij}^2(3-2p_{ij}))$$

The expected cost of the coding operation is

$$R_{S1} = N(3R_C + (1 - \sum_i r_i \sum_j p_{ij}^2(3-2p_{ij}))R_A)$$

S2

Probabilities

The probabilities for the different cases are

$$P\{2\text{-}0|\theta=i\} = p_{ij}^2$$

$$P\{2\text{-}1|\theta=i\} = 2p_{ij}^2 (1-p_{ij})$$

$$P\{1\text{-}1\text{-}1|\theta=i\} = 1 - \sum_j p_{ij}^2(3-2\,p_{ij}) \text{ (the same as in S1)}$$

Thus

$$P\{z=j|\theta=i\} = P\{(2-0)\to j|\theta=i\} + P\{(2-1)\to j|\theta=i\} + \\ + P\{a=j|\theta=i\}\, P\{1-1-1|\theta=i\} \\ = p_{ij}^2 + 2p_{ij}^2(1-p_{ij}) + t_{ij}(1-\sum_j p_{ij}^2(3-2p_{ij}))$$

and we get

$$P\{\text{the outgoing code number is true}\} = \\ = \sum_{i=1}^{K} r_i\, P(z=i|\theta=i) \\ = \sum_i r_i \{3p_{ii}^2-2p_{ii}^3 + t_{ii}(1-\Sigma p_{ij}^2(3-2p_{ij}))\} \\ = \sum_i r_i\, t_{ii} - \sum_i r_i \{t_{ii} \sum_j p_{ij}^2(3-2p_{ij}) - p_{ii}^2(3-2p_{ii})\}$$

and

$$P(z=\theta|t_{ii}=1) = 1-\sum_i r_i \sum_{j\neq i} p_{ij}^2(3-2p_{ij})$$

Costs

The coding is carried out at three stations. At the first statio all elements are coded by two C-coders. Thus 2N elements are coded there. At the second station N'_{S2} elements are coded by C_3 and the expected number of N'_{S2} is

$$E(N'_{S2}) = N\sum_i r_i\, (1-\Sigma p_{ij}^2)$$

At the third coding station N''_{S2} elements are coded by the A-coder. The expected number of N''_{S2} is

$$E(N''_{S2}) = E(N'_{S1}) = N(1-\sum_i r_i \Sigma p_{ij}^2(3-2p_{ij}))$$

The expected cost of the coding operation is

$$R_{S2} = N\{(2 + \sum_i r_i (1 - \sum_j p_{ij}^2))R_C +$$
$$+ (1 - \sum_i r_i \sum_j p_{ij}^2 (3-2\,p_{ij}))R_A\}$$
$$= N\{(3 - \sum_i r_i \sum_j p_{ij}^2)R_C + (1 - \sum_i r_i \sum_j p_{ij}^2 (3-2\,p_{ij}))R_A\}$$

S3

Probabilities

The probabilities for the different cases are

$$P\{2\text{-}0 \mid \theta=i\} = p_{ij}^2$$
$$P\{1\text{-}1 \mid \theta=i\} = (1-\sum_j p_{ij})\, q_{ij}$$

Thus

$$P\{z=j \mid \theta=i\} = P(2\text{-}0 \mid \theta=i)$$
$$+ P(b=j \mid \theta=i)P(1\text{-}1 \mid \theta=i)$$
$$= p_{ij}^2 + q_{ij}(1-\sum_j p_{ij}^2)$$

and we get

$$P\{\text{the outgoing code number is true}\} =$$
$$= \sum_{i=1}^{K} r_i\, P(z=i \mid \theta=i)$$
$$= \sum_i r_i \{p_{ii}^2 + q_{ii}(1-\sum_j p_{ij}^2)\}$$
$$= \sum_i r_i\, q_{ii} - \sum_i r_i \{q_{ii} \sum_j p_{ij}^2 - p_{ii}^2\}$$

and if, for some reason, $q_{ii} = 1$ we get

$$p(z=\theta \mid q_{ii}=1) = 1-\sum_i r_i \sum_{j\neq i} p_{ij}^2$$

Costs

The coding is carried out at two stations. At the first station all elements are coded by two C-coders. Thus 2N elements are coded there. At the second station N'_{S3} elements are coded by B and the expected number of N'_{S3} is

$$E(N'_{S3}) = N\sum_i r_i(1-\sum_j p_{ij}^2) = N(1-\sum_i r_i \sum_j p_{ij}^2)$$

The expected cost of the coding operation is

$$R_{S3} = N\{2R_C + (1-\sum_i r_i \sum_j p_{ij}^2)R_B\}$$

S4

Probabilities

The probabilities for the different cases are

$$P\{2\text{-}0|\theta=i\} = p_{ij}^2$$

$$P\{2\text{-}1|\theta=i\} = 2\, q_{ij}\, p_{ij}(1-p_{ij})$$

$$P\{1\text{-}1\text{-}1|\theta=i\} = t_{ij}\{1-\sum_j(p_{ij}^2 + 2q_{ij}\, p_{ij}(1-p_{ij}))\}$$

Thus

$$P\{z=j|\theta=i\} = p_{ij}^2 + 2\, q_{ij}\, p_{ij}(1-p_{ij}) + t_{ij}\{1-\sum_j(p_{ij}^2 + 2q_{ij}\, p_{ij}(1-p_{ij}))\}$$

and we get

P {the outgoing code number is true} =

$$= \sum_{i=1}^{K} r_i \, P(z=i|\theta=i)$$

$$= \sum_i r_i \, t_{ii} - \sum_i r_i \{t_{ii} \sum_j p_{ij}^2 - p_{ii}^2 +$$

$$+ 2t_{ii} \sum_j q_{ij} \, p_{ij}(1-p_{ij}) - 2 \, q_{ii} \, p_{ii}(1-p_{ii})\}$$

and if $t_{ii} = 1$ we get

$$P(z=\theta|t_{ii}=1) = 1 - \sum_i r_i \sum_{j \neq i} p_{ij}(p_{ij} + 2q_{ij}(1-p_{ij}))$$

Costs

The coding is carried out at three stations. At the first station all elements are coded by two C-coders. Thus 2N elements are coded there. At the second station N'_{S4} elements are coded by B and the expected number of N'_{S4} is

$$E(N'_{S4}) = N(1 - \sum_i r_i \sum_j p_{ij}^2)$$

At the third station N''_{S4} elements are coded by A. The expected number of N''_{S4} is

$$E(N''_{S4}) = N(1 - \sum_i r_i \sum_j p_{ij}(p_{ij} + 2q_{ij}(1 - p_{ij})))$$

The expected cost of the coding operation is

$$R_{S4} = N \, [2R_C + (1 - \sum_i r_i \sum_j p_{ij}^2)R_B$$

$$+(1 - \sum_i r_i \sum_j p_{ij}(p_{ij} + 2q_{ij}(1 - p_{ij})))R_A]$$

S5

Probabilities

The probabilities for the different cases are

$$P\{3\text{-}0|\theta=i\} = p_{ij}^3$$

$$P\{2\text{-}1 \text{ or } 1\text{-}1\text{-}1|\theta=i\} = 1 - \sum_j p_{ij}^3$$

Thus

$$P(z=j|\theta=i) = p_{ij}^3 + q_{ij} \ (1 - \sum_j p_{ij}^3)$$

and we get

P {the outgoing code number is true} =

$$= \sum_{i=1}^{K} r_i \ P(z=i|\theta=i) = \sum_i r_i \ (p_{ii}^3 + q_{ii} \ (1 - \sum_j p_{ij}^3))$$

$$= \sum_i r_i \ q_{ii} - \sum_i r_i \ [q_{ii} \sum_j p_{ij}^3 - p_{ii}^3]$$

Costs

The coding is carried out at two stations. At the first station all elements are coded by three C-coders. Thus 3N elements are coded there. At the second station N'_{S5} elements are coded by B and the expected number of N'_{S5} is

$$E(N'_{S5}) = N \ (1 - \sum_i r_i \sum_j p_{ij}^3)$$

The expected cost of the coding operation is

$$R_{S5} = N(3R_C + (1 - \sum_i r_i \sum_j p_{ij}^3)R_B)$$

S6

Probabilities

The probabilities for the different cases are

$$P\{4\text{-}0|\theta=i\} = p_{ij}^4$$

$$P\{3\text{-}1, \ 2\text{-}2, \ 2\text{-}1\text{-}1, \ 1\text{-}1\text{-}1\text{-}1|\theta=i\} = 1 - \sum_j p_{ij}^4$$

Thus

$$P(z=j|\theta=i) = p_{ij}^4 + q_{ij}\ (1 - \sum_j p_{ij}^4)$$

and we get

P {the outgoing code number is true} =

$$= \sum_{i=1}^{K} r_i\ P(z=i|\theta=i) = \sum_i r_i\ (p_{ii}^4 + q_{ii}(1 - \sum_j p_{ij}^4))$$

$$= \sum_i r_i\ q_{ii} - \sum_i r_i\{q_{ii} \sum_j p_{ij}^4 - p_{ii}^4\}$$

Costs

The coding is carried out at two stations. At the first station all elements are coded by four C-coders. Thus 4N elements are coded there. At the second station N'_{S6} elements are coded by B and the expected number of N'_{S6} is

$$E(N'_{S6}) = N(1 - \sum_i r_i \sum_j p_{ij}^4)$$

The expected cost of the coding operation is

$$R_{S6} = N(4R_C + (1 - \sum_i r_i \sum_j p_{ij}^4)R_B)$$

S7

Probabilities

The probabilities for the different cases are

$$P\{4\text{-}0|\theta=i\} = p_{ij}^4$$

$$P\{3\text{-}1|\theta=i\} = 4\ p_{ij}^3(1-p_{ij})$$

$$P\{2\text{-}1\text{-}1|\theta=i\} = 6\, p_{ij}^2 \sum_{l\neq j} p_{il} \sum_{v\neq l,j} p_{iv}$$

$$= 6\, p_{ij}^2 \,(1-2\, p_{ij} + 2\, p_{ij}^2 - \sum_j p_{ij}^2)$$

$$P\{2\text{-}2|\theta=i\} = 3\sum_j p_{ij}^2 \sum_{l\neq j} p_{il}^2 =$$

$$3\sum_j p_{ij}^2 \{\sum_j p_{ij}^2 - p_{ij}^2\}$$

$$P\{1\text{-}1\text{-}1\text{-}1|\theta=i\} = \sum_j p_{ij} \sum_{l\neq j} p_{il} \sum_{v\neq l,j} p_{iv} \sum_{s\neq l,j,v} p_{is}$$

$$= \sum_j p_{ij} \sum_{l\neq j} p_{il} \sum_{v\neq l,j} p_{iv}\{1-p_{iv}-p_{il}-p_{ij}\}$$

$$= \sum_j p_{ij} \sum_{l\neq j} p_{il} [\sum_v p_{iv}\,(1-p_{iv}-p_{il}-p_{ij})-$$

$$- p_{il}(1-2\, p_{il}-p_{ij}) - p_{ij}(1-p_{il}-2\, p_{ij})]$$

$$= \sum_j p_{ij} \sum_{l\neq j} p_{il} [1 - \sum_v p_{iv}^2 - p_{il} - p_{ij} - p_{il} - 2\, p_{il}^2 +$$

$$+ p_{il}p_{ij} - p_{ij} + p_{il}p_{ij} + 2\, p_{ij}^2]$$

$$= \sum_j p_{ij} \sum_{l\neq j} p_{il} [2\, p_{il}^2 - 2\, p_{il} + 2\, p_{il}\, p_{ij} + 2\, p_{ij}^2 - 2\, p_{ij} + 1 - \sum_v p_{iv}^2]$$

$$= \sum_j p_{ij}[\sum_l p_{il} [2\, p_{il}^2 - 2\, p_{il} + 2\, p_{il}\, p_{ij} +$$

$$+ 2\, p_{ij}^2 - 2\, p_{ij} + 1 - \sum_v p_{iv}^2]$$

$$- p_{ij} [2\, p_{ij}^2 - 2\, p_{ij} + 2\, p_{ij}^2 + 2\, p_{ij}^2 - 2\, p_{ij} +$$

$$+ 1 - \sum_v p_{iv}^2]]$$

$$= \sum_j p_{ij} [2 \sum_l p_{il}^3 - 2 \sum_l p_{il}^2 + 2\, p_{ij} \sum_l p_{il}^2 + 2\, p_{ij}^2$$

$$- 2\, p_{ij} + 1 - \sum_v p_{iv}^2 - 2\, p_{ij}^3 + 2\, p_{ij}^2$$

$$- 2\, p_{ij}^3 - 2\, p_{ij}^3 + 2\, p_{ij}^2 - p_{ij} + p_{ij} \sum_v p_{iv}^2]$$

$$= \sum_j p_{ij} \left[2 \sum_l p_{il}^3 - 2 \sum_l p_{il}^2 + 3 p_{ij} \sum_v p_{iv}^2 + 6 p_{ij}^2 \right.$$

$$\left. - 3 p_{ij} - 6 p_{ij}^3 + 1 - \sum_v p_{iv}^2 \right]$$

$$= 1 - \sum_v p_{iv}^2 + 2 \sum_j p_{ij}^3 + 3 \left(\sum_j p_{ij}^2\right)^2 + 6 \sum_j p_{ij}^3$$

$$- 3 \sum_j p_{ij}^2 - 6 \sum_j p_{ij}^4 - 2 \sum_l p_{il}^2$$

$$= 1 + 3 \left(\sum_j p_{ij}^2\right)^2 + 8 \sum_j p_{ij}^3 - 6 \sum_j p_{ij}^4 - 6 \sum_j p_{ij}^2$$

Thus

$$P(z=j|\theta=i) = p_{ij}^4 + 4 p_{ij}^3 (1-p_{ij})$$

$$+ 6 p_{ij}^2 (1-2 p_{ij} + 2 p_{ij}^2 - \sum_j p_{ij}^2)$$

$$+ q_{ij} \left[1-9 \sum_j p_{ij}^4 + 8 \sum_j p_{ij}^3 - 6 \sum_j p_{ij}^2 + 6 \left(\sum_j p_{ij}^2\right)^2\right]$$

The last of these four terms is the sum of q_{ij} $P\{2\text{-}2|\theta=i\}$ and q_{ij} $P\{1\text{-}1\text{-}1\text{-}1|\theta=i\}$.

We get

P {the outgoing code number is true} =

$$= \sum_{i=1}^{K} r_i \, P(z=i|\theta=i) =$$

$$= \sum_i r_i p_{ii}^4 + 4 \sum_i r_i p_{ii}^3 (1-p_{ii})$$

$$+ 6 \sum_j r_i p_{ii}^2 (1-2 p_{ii} + 2 p_{ii}^2 - \sum_j p_{ij}^2)$$

$$+ \sum_i r_i q_{ii} \left[1-9 \sum_j p_{ij}^4 + 8 \sum_j p_{ij}^3 - 6 \sum_j p_{ij}^2 + 6 \left(\sum_j p_{ij}^2\right)^2\right] .$$

Costs

The coding is carried out at two stations. At the first station all elements are coded by four C-coders. Thus 4N elements are coded there. At the second station N'_{S7} elements are coded by B and the expected number of N'_{S7} is

$$E(N'_{S7}) = N\sum_i r_i \left[6\left(\sum_j p_{ij}^2\right)^2 - 9\sum_j p_{ij}^4 - 6\sum_j p_{ij}^2 + 8\sum_j p_{ij}^3 + 1\right]$$

The expected cost of the coding operation is

$$R_{S7} = 4\ N\ R_C + E(N'_{S7})R_B$$

APPENDIX II

COMPARING SCHEMES

Table 12.5 Probabilities and costs for schemes S1 and S2

NUMBER OF CATEGORIES, K= 2

P	P(OUTGOING CODE IS TRUE) T= .96	T= .98	T=1.00	COST SET I S1	S2	COST SET II S1	S2	COST SET III S1	S2
.80	.8960	.8960	.8960	15000	11600	9000	6960	9000	6960
.82	.9145	.9145	.9145	15000	11476	9000	6886	9000	6886
.84	.9314	.9314	.9314	15000	11344	9000	6806	9000	6806
.86	.9467	.9467	.9467	15000	11204	9000	6722	9000	6722
.88	.9603	.9603	.9603	15000	11056	9000	6634	9000	6634
.90	.9720	.9720	.9720	15000	10900	9000	6540	9000	6540
.92	.9818	.9818	.9818	15000	10736	9000	6442	9000	6442
.94	.9896	.9896	.9896	15000	10564	9000	6338	9000	6338
.96	.9953	.9953	.9953	15000	10384	9000	6230	9000	6230
.98	.9988	.9988	.9988	15000	10196	9000	6118	9000	6118
1.00	1.0000	1.0000	1.0000	15000	10000	9000	6000	9000	6000

NUMBER OF CATEGORIES, K= 5

P	P(OUTGOING CODE IS TRUE) T= .96	T= .98	T=1.00	COST SET I S1	S2	COST SET II S1	S2	COST SET III S1	S2
.80	.9680	.9695	.9710	15375	12125	9450	7500	9675	7725
.82	.9740	.9752	.9764	15310	11907	9372	7330	9558	7516
.84	.9793	.9803	.9813	15250	11690	9300	7164	9449	7313
.86	.9841	.9849	.9856	15195	11472	9234	7000	9351	7117
.88	.9882	.9888	.9894	15146	11256	9175	6841	9262	6928
.90	.9918	.9922	.9926	15103	11041	9124	6686	9186	6748
.92	.9947	.9950	.9953	15067	10827	9081	6537	9121	6577
.94	.9970	.9972	.9973	15038	10616	9046	6393	9069	6416
.96	.9987	.9987	.9988	15017	10407	9021	6255	9031	6265
.98	.9997	.9997	.9997	15004	10202	9005	6124	9008	6126
1.00	1.0000	1.0000	1.0000	15000	10000	9000	6000	9000	6000

NUMBER OF CATEGORIES, K= 10

P	P(OUTGOING CODE IS TRUE) T= .96	T= .98	T=1.00	COST SET I S1	S2	COST SET II S1	S2	COST SET III S1	S2
.80	.9832	.9850	.9869	15454	12232	9545	7612	9818	7884
.82	.9863	.9878	.9893	15374	11994	9449	7421	9674	7646
.84	.9892	.9904	.9916	15301	11759	9361	7236	9542	7416
.86	.9917	.9926	.9935	15234	11525	9281	7056	9422	7196
.88	.9938	.9945	.9952	15175	11295	9210	6882	9315	6987
.90	.9957	.9962	.9967	15123	11068	9148	6715	9222	6789
.92	.9972	.9976	.9979	15080	10845	9096	6555	9144	6603
.94	.9984	.9986	.9988	15046	10626	9055	6403	9083	6431
.96	.9993	.9994	.9995	15021	10412	9025	6260	9037	6272
.98	.9998	.9998	.9999	15005	10203	9006	6125	9009	6128
1.00	1.0000	1.0000	1.0000	15000	10000	9000	6000	9000	6000

NUMBER OF CATEGORIES, K= 50

P	P(OUTGOING CODE IS TRUE) T= .96	T= .98	T=1.00	COST SET I S1	S2	COST SET II S1	S2	COST SET III S1	S2
.80	.9935	.9955	.9976	15508	12304	9609	7687	9914	7992
.82	.9947	.9964	.9980	15418	12052	9501	7482	9752	7733
.84	.9958	.9971	.9984	15335	11805	9402	7284	9603	7485
.86	.9967	.9978	.9988	15261	11561	9313	7093	9469	7249
.88	.9976	.9983	.9991	15194	11321	9233	6909	9350	7026
.90	.9983	.9988	.9994	15137	11086	9164	6734	9246	6816
.92	.9989	.9993	.9996	15089	10856	9107	6567	9160	6620
.94	.9994	.9996	.9998	15051	10632	9061	6410	9091	6440
.96	.9997	.9998	.9999	15023	10415	9027	6263	9041	6276
.98	.9999	1.0000	1.0000	15006	10204	9007	6126	9010	6129
1.00	1.0000	1.0000	1.0000	15000	10000	9000	6000	9000	6000

Table 12.6 Probabilities and costs for scheme S3.

NUMBER OF CATEGORIES, K= 2

P	P(OUTGOING CODE IS TRUE) Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9280	.9344	.9408	.9472	.9536	.9600	11600	7280	7920
.82	.9381	.9440	.9499	.9558	.9617	.9676	11476	7181	7771
.84	.9475	.9529	.9583	.9636	.9690	.9744	11344	7075	7613
.86	.9563	.9611	.9660	.9708	.9756	.9804	11204	6963	7445
.88	.9645	.9687	.9729	.9772	.9814	.9856	11056	6845	7267
.90	.9720	.9756	.9792	.9828	.9864	.9900	10900	6720	7080
.92	.9789	.9818	.9848	.9877	.9907	.9936	10736	6589	6883
.94	.9851	.9874	.9896	.9919	.9941	.9964	10564	6451	6677
.96	.9907	.9923	.9938	.9953	.9969	.9984	10384	6307	6461
.98	.9957	.9965	.9972	.9980	.9988	.9996	10196	6157	6235
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	10000	6000	6000

NUMBER OF CATEGORIES, K= 5

P	P(OUTGOING CODE IS TRUE) Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9550	.9620	.9690	.9760	.9830	.9900	11750	7400	8100
.82	.9600	.9663	.9727	.9791	.9855	.9919	11598	7278	7917
.84	.9648	.9706	.9763	.9821	.9878	.9936	11440	7152	7728
.86	.9696	.9747	.9798	.9849	.9900	.9951	11277	7022	7533
.88	.9742	.9786	.9831	.9875	.9920	.9964	11110	6888	7332
.90	.9788	.9825	.9863	.9900	.9938	.9975	10938	6750	7125
.92	.9832	.9862	.9893	.9923	.9954	.9984	10760	6608	6912
.94	.9876	.9899	.9922	.9945	.9968	.9991	10577	6462	6693
.96	.9918	.9934	.9949	.9965	.9980	.9996	10390	6312	6468
.98	.9960	.9967	.9975	.9983	.9991	.9999	10197	6158	6237
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	10000	6000	6000

NUMBER OF CATEGORIES, K= 10

P	P(OUTGOING CODE IS TRUE) Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9600	.9671	.9742	.9813	.9884	.9956	11778	7422	8133
.82	.9640	.9705	.9770	.9834	.9899	.9964	11620	7296	7944
.84	.9680	.9738	.9797	.9855	.9913	.9972	11458	7166	7749
.86	.9720	.9772	.9823	.9875	.9927	.9978	11291	7033	7549
.88	.9760	.9805	.9850	.9894	.9939	.9984	11120	6896	7344
.90	.9800	.9838	.9876	.9913	.9951	.9989	10944	6756	7133
.92	.9840	.9871	.9901	.9932	.9962	.9993	10764	6612	6917
.94	.9880	.9903	.9926	.9950	.9973	.9996	10580	6464	6696
.96	.9920	.9936	.9951	.9967	.9983	.9998	10391	6313	6469
.98	.9960	.9968	.9976	.9984	.9992	1.0000	10198	6158	6237
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	10000	6000	6000

NUMBER OF CATEGORIES, K= 50

P	P(OUTGOING CODE IS TRUE) Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9633	.9704	.9776	.9848	.9920	.9992	11796	7437	8155
.82	.9666	.9732	.9797	.9863	.9928	.9993	11635	7308	7962
.84	.9701	.9760	.9818	.9877	.9936	.9995	11469	7176	7763
.86	.9736	.9788	.9840	.9892	.9944	.9996	11300	7040	7560
.88	.9772	.9817	.9862	.9907	.9952	.9997	11127	6901	7352
.90	.9808	.9846	.9884	.9922	.9960	.9998	10949	6759	7139
.92	.9845	.9876	.9907	.9937	.9968	.9999	10767	6614	6921
.94	.9883	.9906	.9929	.9953	.9976	.9999	10582	6465	6698
.96	.9921	.9937	.9953	.9968	.9984	1.0000	10392	6313	6470
.98	.9960	.9968	.9976	.9984	.9992	1.0000	10198	6158	6238
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	10000	6000	6000

Table 12.7 a Probabilities for scheme S4.

NUMBER OF CATEGORIES, K= 2

P	P(THE OUTGOING CODE IS TRUE) Q= .90			Q= .92			Q= .94		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9280	.9280	.9280	.9344	.9344	.9344	.9408	.9408	.9408
.82	.9381	.9381	.9381	.9440	.9440	.9440	.9499	.9499	.9499
.84	.9475	.9475	.9475	.9529	.9529	.9529	.9583	.9583	.9583
.86	.9563	.9563	.9563	.9611	.9611	.9611	.9660	.9660	.9660
.88	.9645	.9645	.9645	.9687	.9687	.9687	.9729	.9729	.9729
.90	.9720	.9720	.9720	.9756	.9756	.9756	.9792	.9792	.9792
.92	.9789	.9789	.9789	.9818	.9818	.9818	.9848	.9848	.9848
.94	.9851	.9851	.9851	.9874	.9874	.9874	.9896	.9896	.9896
.96	.9907	.9907	.9907	.9923	.9923	.9923	.9938	.9938	.9938
.98	.9957	.9957	.9957	.9965	.9965	.9965	.9972	.9972	.9972
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

P	P(THE OUTGOING CODE IS TRUE) Q= .96			Q= .98			Q=1.00		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9472	.9472	.9472	.9536	.9536	.9536	.9600	.9600	.9600
.82	.9558	.9558	.9558	.9617	.9617	.9617	.9676	.9676	.9676
.84	.9636	.9636	.9636	.9690	.9690	.9690	.9744	.9744	.9744
.86	.9708	.9708	.9708	.9756	.9756	.9756	.9804	.9804	.9804
.88	.9772	.9772	.9772	.9814	.9814	.9814	.9856	.9856	.9856
.90	.9828	.9828	.9828	.9864	.9864	.9864	.9900	.9900	.9900
.92	.9877	.9877	.9877	.9907	.9907	.9907	.9936	.9936	.9936
.94	.9919	.9919	.9919	.9941	.9941	.9941	.9964	.9964	.9964
.96	.9953	.9953	.9953	.9969	.9969	.9969	.9984	.9984	.9984
.98	.9980	.9980	.9980	.9988	.9988	.9988	.9996	.9996	.9996
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

NUMBER OF CATEGORIES, K= 5

P	P(THE OUTGOING CODE IS TRUE) Q= .90			Q= .92			Q= .94		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9784	.9795	.9805	.9805	.9814	.9824	.9826	.9834	.9843
.82	.9815	.9824	.9833	.9834	.9842	.9850	.9853	.9860	.9867
.84	.9844	.9852	.9859	.9861	.9868	.9875	.9878	.9884	.9890
.86	.9871	.9877	.9883	.9886	.9891	.9897	.9900	.9905	.9910
.88	.9895	.9901	.9906	.9908	.9913	.9917	.9921	.9925	.9929
.90	.9918	.9922	.9926	.9929	.9932	.9936	.9940	.9943	.9946
.92	.9939	.9942	.9945	.9947	.9950	.9953	.9956	.9958	.9960
.94	.9957	.9959	.9961	.9964	.9965	.9967	.9970	.9972	.9973
.96	.9973	.9975	.9976	.9978	.9979	.9980	.9982	.9983	.9984
.98	.9988	.9988	.9989	.9990	.9991	.9991	.9992	.9993	.9993
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

P	P(THE OUTGOING CODE IS TRUE) Q= .96			Q= .98			Q=1.00		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9846	.9854	.9862	.9867	.9874	.9881	.9888	.9894	.9900
.82	.9872	.9878	.9885	.9890	.9896	.9902	.9909	.9914	.9919
.84	.9895	.9900	.9905	.9911	.9916	.9921	.9928	.9932	.9936
.86	.9915	.9920	.9924	.9930	.9934	.9937	.9945	.9948	.9951
.88	.9934	.9937	.9941	.9947	.9950	.9952	.9960	.9962	.9964
.90	.9950	.9953	.9956	.9961	.9963	.9965	.9972	.9974	.9975
.92	.9965	.9966	.9968	.9973	.9975	.9976	.9982	.9983	.9984
.94	.9977	.9978	.9979	.9983	.9984	.9985	.9990	.9990	.9991
.96	.9987	.9987	.9988	.9991	.9992	.9992	.9996	.9996	.9996
.98	.9994	.9995	.9995	.9997	.9997	.9997	.9999	.9999	.9999
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Table 12.7 a (continued).

NUMBER OF CATEGORIES, K= 10

P	P(THE OUTGOING CODE IS TRUE) Q= .90			Q= .92			Q= .94		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9887	.9899	.9912	.9898	.9909	.9921	.9909	.9919	.9929
.82	.9903	.9914	.9925	.9913	.9923	.9933	.9923	.9932	.9940
.84	.9918	.9927	.9937	.9927	.9935	.9944	.9936	.9943	.9951
.86	.9932	.9940	.9948	.9940	.9947	.9954	.9948	.9954	.9960
.88	.9945	.9951	.9958	.9952	.9957	.9963	.9959	.9963	.9968
.90	.9957	.9962	.9967	.9963	.9967	.9971	.9968	.9972	.9976
.92	.9968	.9972	.9975	.9972	.9976	.9979	.9977	.9980	.9982
.94	.9977	.9980	.9983	.9981	.9983	.9985	.9984	.9986	.9988
.96	.9986	.9988	.9989	.9988	.9990	.9991	.9991	.9992	.9993
.98	.9994	.9994	.9995	.9995	.9995	.9996	.9996	.9996	.9997
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

P	P(THE OUTGOING CODE IS TRUE) Q= .96			Q= .98			Q=1.00		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9920	.9929	.9938	.9930	.9939	.9947	.9941	.9948	.9956
.82	.9933	.9941	.9948	.9943	.9949	.9956	.9952	.9958	.9964
.84	.9945	.9951	.9958	.9954	.9959	.9965	.9962	.9967	.9972
.86	.9956	.9961	.9966	.9963	.9968	.9972	.9971	.9975	.9978
.88	.9965	.9969	.9973	.9972	.9975	.9979	.9979	.9981	.9984
.90	.9974	.9977	.9980	.9980	.9982	.9984	.9985	.9987	.9989
.92	.9981	.9984	.9986	.9986	.9988	.9989	.9991	.9992	.9993
.94	.9988	.9989	.9991	.9991	.9992	.9993	.9995	.9995	.9996
.96	.9993	.9994	.9995	.9995	.9996	.9996	.9998	.9998	.9998
.98	.9997	.9997	.9998	.9998	.9998	.9999	.9999	.9999	1.0000
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

NUMBER OF CATEGORIES, K= 50

P	P(THE OUTGOING CODE IS TRUE) Q= .90			Q= .92			Q= .94		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9956	.9970	.9984	.9960	.9973	.9985	.9964	.9975	.9987
.82	.9962	.9974	.9986	.9966	.9977	.9988	.9969	.9979	.9989
.84	.9968	.9978	.9988	.9971	.9980	.9990	.9975	.9983	.9991
.86	.9973	.9982	.9990	.9976	.9984	.9991	.9979	.9986	.9993
.88	.9978	.9985	.9992	.9981	.9987	.9993	.9984	.9989	.9994
.90	.9983	.9988	.9994	.9985	.9990	.9995	.9987	.9991	.9996
.92	.9987	.9991	.9995	.9989	.9993	.9996	.9991	.9994	.9997
.94	.9991	.9994	.9997	.9992	.9995	.9997	.9994	.9996	.9998
.96	.9994	.9996	.9998	.9995	.9997	.9998	.9996	.9997	.9999
.98	.9997	.9998	.9999	.9998	.9999	.9999	.9998	.9999	.9999
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

P	P(THE OUTGOING CODE IS TRUE) Q= .96			Q= .98			Q=1.00		
	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00	T= .96	T= .98	T=1.00
.80	.9968	.9978	.9989	.9972	.9981	.9990	.9976	.9984	.9992
.82	.9973	.9982	.9990	.9977	.9984	.9992	.9981	.9987	.9993
.84	.9978	.9985	.9992	.9981	.9987	.9993	.9985	.9990	.9995
.86	.9982	.9988	.9994	.9985	.9990	.9995	.9988	.9992	.9996
.88	.9986	.9991	.9995	.9989	.9992	.9996	.9991	.9994	.9997
.90	.9990	.9993	.9996	.9992	.9994	.9997	.9994	.9996	.9998
.92	.9993	.9995	.9997	.9994	.9996	.9998	.9996	.9997	.9999
.94	.9995	.9997	.9998	.9996	.9998	.9999	.9998	.9999	.9999
.96	.9997	.9998	.9999	.9998	.9999	.9999	.9999	.9999	1.0000
.98	.9999	.9999	1.0000	.9999	1.0000	1.0000	1.0000	1.0000	1.0000
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Table 12.7 b Costs for scheme S4.

NUMBER OF CATEGORIES, K= 2

P	COSTS FOR DIFFERENT COST SETS								
	Q= .90			Q= .92			Q= .94		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	11600	7280	7920	11600	7280	7920	11600	7280	7920
.82	11476	7181	7771	11476	7181	7771	11476	7181	7771
.84	11344	7075	7613	11344	7075	7613	11344	7075	7613
.86	11204	6963	7445	11204	6963	7445	11204	6963	7445
.88	11056	6845	7267	11056	6845	7267	11056	6845	7267
.90	10900	6720	7080	10900	6720	7080	10900	6720	7080
.92	10736	6589	6883	10736	6589	6883	10736	6589	6883
.94	10564	6451	6677	10564	6451	6677	10564	6451	6677
.96	10384	6307	6461	10384	6307	6461	10384	6307	6461
.98	10196	6157	6235	10196	6157	6235	10196	6157	6235
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

P	COSTS FOR DIFFERENT COST SETS								
	Q= .96			Q= .98			Q=1.00		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	11600	7280	7920	11600	7280	7920	11600	7280	7920
.82	11476	7181	7771	11476	7181	7771	11476	7181	7771
.84	11344	7075	7613	11344	7075	7613	11344	7075	7613
.86	11204	6963	7445	11204	6963	7445	11204	6963	7445
.88	11056	6845	7267	11056	6845	7267	11056	6845	7267
.90	10900	6720	7080	10900	6720	7080	10900	6720	7080
.92	10736	6589	6883	10736	6589	6883	10736	6589	6883
.94	10564	6451	6677	10564	6451	6677	10564	6451	6677
.96	10384	6307	6461	10384	6307	6461	10384	6307	6461
.98	10196	6157	6235	10196	6157	6235	10196	6157	6235
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

NUMBER OF CATEGORIES, K= 5

P	COSTS FOR DIFFERENT COST SETS								
	Q= .90			Q= .92			Q= .94		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	12013	7715	8573	11990	7688	8532	11967	7661	8492
.82	11824	7549	8324	11803	7524	8286	11782	7499	8249
.84	11632	7382	8074	11613	7359	8039	11594	7336	8004
.86	11438	7214	7821	11420	7193	7790	11403	7173	7759
.88	11241	7045	7567	11225	7026	7539	11210	7008	7512
.90	11041	6874	7311	11028	6858	7287	11014	6842	7263
.92	10838	6702	7052	10827	6689	7033	10816	6676	7014
.94	10633	6528	6792	10624	6518	6777	10616	6508	6762
.96	10424	6353	6530	10419	6347	6520	10413	6340	6510
.98	10214	6177	6266	10211	6174	6261	10208	6170	6255
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

P	COSTS FOR DIFFERENT COST SETS								
	Q= .96			Q= .98			Q=1.00		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	11945	7634	8451	11923	7607	8411	11900	7580	8370
.82	11761	7474	8211	11740	7449	8173	11719	7424	8136
.84	11574	7313	7970	11555	7290	7935	11536	7267	7901
.86	11386	7152	7728	11368	7131	7696	11351	7110	7665
.88	11195	6990	7484	11179	6971	7457	11164	6953	7429
.90	11001	6827	7240	10988	6811	7216	10975	6795	7193
.92	10806	6663	6994	10795	6650	6975	10784	6637	6955
.94	10608	6498	6747	10599	6488	6732	10591	6478	6717
.96	10407	6333	6499	10402	6326	6489	10396	6319	6479
.98	10205	6167	6250	10202	6163	6245	10199	6160	6240
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

Table 12.7 b (continued).

NUMBER OF CATEGORIES, K= 10

P	COSTS FOR DIFFERENT COST SETS								
	Q= .90			Q= .92			Q= .94		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	12094	7801	8702	12066	7768	8652	12039	7735	8603
.82	11892	7622	8434	11866	7592	8388	11841	7561	8341
.84	11688	7443	8165	11665	7415	8123	11642	7387	8080
.86	11483	7264	7895	11462	7238	7857	11441	7213	7820
.88	11276	7084	7626	11258	7062	7592	11239	7039	7559
.90	11068	6904	7356	11052	6885	7327	11036	6866	7299
.92	10858	6723	7085	10845	6708	7062	10832	6692	7039
.94	10646	6543	6814	10636	6531	6796	10626	6519	6779
.96	10432	6362	6543	10425	6354	6531	10419	6346	6519
.98	10217	6181	6272	10213	6177	6266	10210	6173	6259
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

P	COSTS FOR DIFFERENT COST SETS								
	Q= .96			Q= .98			Q=1.00		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	12011	7702	8553	11983	7669	8503	11956	7636	8453
.82	11815	7530	8295	11790	7500	8249	11764	7469	8203
.84	11618	7359	8038	11595	7331	7996	11572	7303	7954
.86	11420	7188	7782	11399	7163	7744	11378	7137	7706
.88	11221	7017	7526	11202	6995	7492	11184	6973	7459
.90	11020	6847	7270	11005	6828	7242	10989	6809	7213
.92	10819	6677	7015	10806	6661	6992	10793	6646	6969
.94	10616	6507	6761	10606	6495	6743	10596	6483	6725
.96	10412	6338	6507	10405	6330	6494	10398	6321	6482
.98	10207	6169	6253	10203	6165	6247	10200	6160	6241
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

NUMBER OF CATEGORIES, K= 50

P	COSTS FOR DIFFERENT COST SETS								
	Q= .90			Q= .92			Q= .94		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	12148	7859	8788	12117	7822	8732	12085	7784	8676
.82	11937	7671	8506	11909	7636	8455	11880	7602	8403
.84	11726	7483	8225	11700	7452	8178	11673	7420	8131
.86	11514	7296	7944	11490	7268	7902	11467	7240	7860
.88	11300	7110	7664	11280	7085	7627	11259	7060	7590
.90	11086	6924	7385	11068	6902	7354	11051	6881	7322
.92	10871	6738	7107	10856	6721	7081	10842	6703	7055
.94	10654	6553	6829	10643	6539	6809	10632	6526	6789
.96	10437	6368	6552	10430	6359	6538	10422	6350	6525
.98	10219	6184	6276	10215	6179	6269	10211	6175	6262
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

P	COSTS FOR DIFFERENT COST SETS								
	Q= .96			Q= .98			Q=1.00		
	SET I	SET II	SET III	SET I	SET II	SET III	SET I	SET II	SET III
.80	12054	7747	8620	12023	7709	8564	11992	7672	8508
.82	11851	7567	8351	11822	7533	8299	11793	7498	8247
.84	11647	7389	8083	11621	7357	8036	11595	7326	7989
.86	11443	7212	7817	11420	7183	7775	11396	7155	7733
.88	11238	7035	7553	11218	7011	7516	11197	6986	7479
.90	11033	6860	7290	11016	6839	7259	10998	6818	7227
.92	10827	6686	7029	10813	6669	7003	10799	6651	6977
.94	10621	6513	6769	10610	6500	6750	10599	6486	6730
.96	10415	6341	6511	10407	6332	6498	10400	6323	6484
.98	10208	6170	6255	10204	6165	6248	10200	6161	6241
1.00	10000	6000	6000	10000	6000	6000	10000	6000	6000

Table 12.8 Probabilities and costs for scheme S5.

NUMBER OF CATEGORIES, K= 2

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9440	.9536	.9632	.9728	.9824	.9920	17400	10920	11880
.82	.9499	.9537	.9676	.9765	.9853	.9942	17214	10771	11657
.84	.9556	.9636	.9717	.9798	.9878	.9959	17016	10613	11419
.86	.9611	.9684	.9756	.9828	.9900	.9973	16806	10445	11167
.88	.9666	.9729	.9793	.9856	.9919	.9983	16584	10267	10901
.90	.9720	.9774	.9828	.9882	.9936	.9990	16350	10080	10620
.92	.9774	.9818	.9862	.9907	.9951	.9995	16104	9883	10325
.94	.9829	.9862	.9896	.9930	.9964	.9998	15846	9677	10015
.96	.9884	.9907	.9930	.9953	.9976	.9999	15576	9461	9691
.98	.9941	.9953	.9965	.9976	.9988	1.0000	15294	9235	9353
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	15000	9000	9000

NUMBER OF CATEGORIES, K= 5

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9508	.9605	.9703	.9800	.9898	.9995	17438	10950	11925
.82	.9548	.9638	.9727	.9817	.9907	.9996	17241	10793	11690
.84	.9590	.9672	.9753	.9835	.9916	.9997	17035	10628	11442
.86	.9635	.9707	.9780	.9853	.9926	.9998	16819	10455	11183
.88	.9681	.9744	.9808	.9872	.9935	.9999	16592	10274	10911
.90	.9728	.9783	.9837	.9891	.9945	.9999	16355	10084	10626
.92	.9778	.9823	.9867	.9911	.9955	1.0000	16106	9885	10328
.94	.9830	.9864	.9898	.9932	.9966	1.0000	15847	9678	10016
.96	.9885	.9908	.9931	.9954	.9977	1.0000	15576	9461	9692
.98	.9941	.9953	.9965	.9976	.9988	1.0000	15294	9235	9353
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	15000	9000	9000

NUMBER OF CATEGORIES, K= 10

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9511	.9609	.9706	.9804	.9901	.9999	17440	10952	11927
.82	.9551	.9640	.9730	.9820	.9910	.9999	17243	10794	11691
.84	.9592	.9674	.9755	.9837	.9918	.9999	17036	10629	11443
.86	.9636	.9709	.9781	.9854	.9927	1.0000	16820	10456	11183
.88	.9681	.9745	.9809	.9872	.9936	1.0000	16593	10274	10911
.90	.9729	.9783	.9837	.9891	.9946	1.0000	16355	10084	10626
.92	.9779	.9823	.9867	.9911	.9956	1.0000	16107	9885	10328
.94	.9831	.9864	.9898	.9932	.9966	1.0000	15847	9678	10016
.96	.9885	.9908	.9931	.9954	.9977	1.0000	15576	9461	9692
.98	.9941	.9953	.9965	.9976	.9988	1.0000	15294	9235	9353
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	15000	9000	9000

NUMBER OF CATEGORIES, K= 50

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9512	.9610	.9707	.9805	.9902	1.0000	17440	10952	11928
.82	.9551	.9641	.9731	.9821	.9910	1.0000	17243	10795	11692
.84	.9593	.9674	.9756	.9837	.9919	1.0000	17036	10629	11444
.86	.9636	.9709	.9782	.9854	.9927	1.0000	16820	10456	11184
.88	.9681	.9745	.9809	.9873	.9936	1.0000	16593	10274	10911
.90	.9729	.9783	.9837	.9892	.9946	1.0000	16355	10084	10626
.92	.9779	.9823	.9867	.9911	.9956	1.0000	16107	9885	10328
.94	.9831	.9864	.9898	.9932	.9966	1.0000	15847	9678	10016
.96	.9885	.9908	.9931	.9954	.9977	1.0000	15576	9461	9692
.98	.9941	.9953	.9965	.9976	.9988	1.0000	15294	9235	9353
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	15000	9000	9000

Table 12.9 Probabilities and costs for scheme S6.

NUMBER OF CATEGORIES, K= 2

P	P(OUTGOING CODE IS TRUE						COSTS		
	Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	I	II	III
.80	.9395	.9513	.9631	.9748	.9866	.9984	22944	14355	15533
.82	.9443	.9552	.9661	.9771	.9880	.9990	22734	14187	15281
.84	.9492	.9592	.9693	.9793	.9893	.9993	22507	14006	15009
.86	.9544	.9634	.9725	.9815	.9906	.9996	22263	13810	14716
.88	.9598	.9678	.9758	.9838	.9918	.9998	22000	13600	14401
.90	.9655	.9724	.9793	.9861	.9930	.9999	21719	13375	14063
.92	.9716	.9773	.9829	.9886	.9943	1.0000	21418	13134	13701
.94	.9781	.9824	.9868	.9912	.9956	1.0000	21096	12877	13315
.96	.9849	.9879	.9910	.9940	.9970	1.0000	20753	12603	12904
.98	.9922	.9938	.9953	.9969	.9984	1.0000	20388	12311	12466
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

NUMBER OF CATEGORIES, K= 5

P	P(OUTGOING CODE IS TRUE						COSTS		
	Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	I	II	III
.80	.9409	.9527	.9646	.9764	.9882	1.0000	22952	14361	15542
.82	.9452	.9562	.9671	.9781	.9890	1.0000	22739	14191	15287
.84	.9498	.9598	.9699	.9799	.9899	1.0000	22511	14008	15013
.86	.9547	.9638	.9728	.9819	.9909	1.0000	22265	13812	14718
.88	.9600	.9680	.9760	.9840	.9920	1.0000	22002	13601	14402
.90	.9656	.9725	.9794	.9862	.9931	1.0000	21719	13376	14063
.92	.9716	.9773	.9830	.9887	.9943	1.0000	21418	13134	13702
.94	.9781	.9825	.9868	.9912	.9956	1.0000	21096	12877	13316
.96	.9849	.9879	.9910	.9940	.9970	1.0000	20753	12603	12904
.98	.9922	.9938	.9953	.9969	.9984	1.0000	20388	12311	12466
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

NUMBER OF CATEGORIES, K= 10

P	P(OUTGOING CODE IS TRUE						COSTS		
	Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	I	II	III
.80	.9410	.9528	.9646	.9764	.9882	1.0000	22952	14362	15542
.82	.9452	.9562	.9671	.9781	.9890	1.0000	22739	14192	15287
.84	.9498	.9598	.9699	.9799	.9900	1.0000	22511	14009	15013
.86	.9547	.9638	.9728	.9819	.9909	1.0000	22265	13812	14718
.88	.9600	.9680	.9760	.9840	.9920	1.0000	22002	13601	14402
.90	.9656	.9725	.9794	.9862	.9931	1.0000	21719	13376	14063
.92	.9716	.9773	.9830	.9887	.9943	1.0000	21418	13134	13702
.94	.9781	.9825	.9868	.9912	.9956	1.0000	21096	12877	13316
.96	.9849	.9879	.9910	.9940	.9970	1.0000	20753	12603	12904
.98	.9922	.9938	.9953	.9969	.9984	1.0000	20388	12311	12466
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

NUMBER OF CATEGORIES, K= 50

P	P(OUTGOING CODE IS TRUE						COSTS		
	Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	I	II	III
.80	.9410	.9528	.9646	.9764	.9882	1.0000	22952	14362	15542
.82	.9452	.9562	.9671	.9781	.9890	1.0000	22739	14192	15287
.84	.9498	.9598	.9699	.9799	.9900	1.0000	22511	14009	15013
.86	.9547	.9638	.9728	.9819	.9909	1.0000	22265	13812	14718
.88	.9600	.9680	.9760	.9840	.9920	1.0000	22002	13601	14402
.90	.9656	.9725	.9794	.9862	.9931	1.0000	21719	13376	14063
.92	.9716	.9773	.9830	.9887	.9943	1.0000	21418	13134	13702
.94	.9781	.9825	.9868	.9912	.9956	1.0000	21096	12877	13316
.96	.9849	.9879	.9910	.9940	.9970	1.0000	20753	12603	12904
.98	.9922	.9938	.9953	.9969	.9984	1.0000	20388	12311	12466
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

Table 12.10 Probabilities and costs for scheme S7.

NUMBER OF CATEGORIES, K= 2

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9574	.9605	.9636	.9667	.9697	.9728	20768	12614	12922
.82	.9667	.9694	.9720	.9746	.9772	.9798	20654	12523	12784
.84	.9747	.9769	.9791	.9812	.9834	.9856	20542	12434	12650
.86	.9815	.9832	.9850	.9867	.9884	.9902	20435	12348	12522
.88	.9870	.9884	.9897	.9910	.9924	.9937	20335	12268	12401
.90	.9914	.9924	.9934	.9944	.9953	.9963	20243	12194	12292
.92	.9948	.9955	.9961	.9968	.9974	.9981	20163	12130	12195
.94	.9973	.9976	.9980	.9984	.9988	.9992	20095	12076	12115
.96	.9989	.9990	.9992	.9994	.9996	.9998	20044	12035	12053
.98	.9997	.9998	.9998	.9999	.9999	1.0000	20012	12009	12014
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

NUMBER OF CATEGORIES, K= 5

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9779	.9789	.9799	.9808	.9818	.9828	20242	12194	12290
.82	.9832	.9840	.9848	.9856	.9864	.9872	20200	12160	12241
.84	.9877	.9883	.9890	.9896	.9902	.9909	20162	12130	12194
.86	.9913	.9918	.9923	.9928	.9933	.9938	20127	12101	12152
.88	.9941	.9945	.9949	.9953	.9957	.9960	20095	12076	12114
.90	.9963	.9966	.9969	.9971	.9974	.9977	20068	12054	12081
.92	.9979	.9981	.9983	.9984	.9986	.9988	20044	12035	12053
.94	.9990	.9991	.9992	.9993	.9994	.9995	20025	12020	12030
.96	.9996	.9997	.9997	.9998	.9998	.9998	20012	12009	12014
.98	.9999	.9999	.9999	1.0000	1.0000	1.0000	20003	12002	12004
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

NUMBER OF CATEGORIES, K= 10

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9877	.9884	.9892	.9899	.9906	.9913	20178	12142	12213
.82	.9907	.9913	.9919	.9924	.9930	.9936	20141	12113	12170
.84	.9932	.9937	.9941	.9945	.9950	.9954	20109	12088	12131
.86	.9953	.9956	.9959	.9962	.9966	.9969	20082	12066	12098
.88	.9968	.9971	.9973	.9975	.9978	.9980	20059	12047	12070
.90	.9980	.9982	.9984	.9985	.9987	.9988	20040	12032	12048
.92	.9989	.9990	.9991	.9992	.9993	.9994	20025	12020	12030
.94	.9995	.9995	.9996	.9996	.9997	.9997	20013	12011	12016
.96	.9998	.9998	.9999	.9999	.9999	.9999	20006	12005	12007
.98	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20001	12001	12002
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

NUMBER OF CATEGORIES, K= 50

P	P(OUTGOING CODE IS TRUE Q= .90	Q= .92	Q= .94	Q= .96	Q= .98	Q=1.00	COSTS I	II	III
.80	.9954	.9960	.9966	.9971	.9977	.9983	20143	12114	12172
.82	.9966	.9970	.9974	.9979	.9983	.9987	20103	12086	12129
.84	.9975	.9978	.9981	.9985	.9988	.9991	20079	12063	12094
.86	.9983	.9985	.9987	.9989	.9992	.9994	20055	12044	12066
.88	.9989	.9990	.9992	.9993	.9995	.9996	20036	12029	12044
.90	.9993	.9994	.9995	.9996	.9997	.9998	20022	12018	12027
.92	.9996	.9997	.9997	.9998	.9998	.9999	20012	12010	12015
.94	.9998	.9999	.9999	.9999	.9999	.9999	20006	12005	12007
.96	.9999	1.0000	1.0000	1.0000	1.0000	1.0000	20002	12002	12002
.98	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000
1.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	20000	12000	12000

13 The extent of verification

Verification can be administered on a total or a sampling basis, and often a mixture between the two is the most suitable. A common approach is to start with total verification and then, when the process has become stable, move over to sampling verification. As pointed out in chapter 6, such schemes as acceptance sampling, process control and hybrid schemes may be used. In this chapter these matters will be discussed in more detail.

13.1 Statistical quality control applications in manual coding

Acceptance sampling was used in the 1965, 1970 and 1975 Swedish Censuses of Population and Housing and also in the US censuses during the last three decades.

During the 1965 Swedish Censuses of Population and Housing rather few variables were coded and dependent verification was used. An acceptance sampling plan with AOQL = 5 % was used throughout the entire operation. This plan was supposed to guarantee that, on the average, at most 5 % of the schedules contained one or more coding errors after control. This scheme, including rectifying inspection on a total basis for subquality lots, resulted in a flow between total and sampling control for individual production coders. As a result rather few errors were detected but, since the drawbacks of dependent verification were not fully recognized at the time, the operation went smoothly.

In designing the coding control procedures for the 1970 Swedish census the situation was different. Independent verification was to be used and the number of variables to code had increased over the previous census. Thus for the variables of economic activity it was decided that a sampling plan with AOQL = 10 %

should be used. Unfortunately this plan was far too optimistic. The process average was too high to begin with, too many (!) errors were found by the effective independent verification and as a result the acceptance sampling scheme had to be abandoned. The resources simply were not available to carry out all the rectifying inspection called for, and the last part of the quality control program was restricted to process control only. In this process control coding error rates were plotted and on-the-spot action was taken when necessary. However, no more lots were rectified after that.

During the 1975 Swedish Census of Population hybrid procedures were used. It was decided that a control volume constraint should be used. Special financial and administrative resources were allocated corresponding to a pre-specified control volume of 25 percent. Thus the efficiency criterion was "best possible quality within budget constraints" and as a consequence values of the parameters n and c could be changed during the production according to the constraints.

In order to protect the producers of data it is important to have means for early detection of coders who do not meet performance standards. The results of an acceptance sampling scheme (or perhaps a hybrid scheme without rectification) could be used to assign points to each coder. Such point systems are discussed in Cook (1959), Minton (1970 a) and Minton and Krivitsky (1971). In such systems a coder who has qualified for production coding is given a bonus of C points. A lot sample of the coder's work is made and after verification it is decided whether or not the lot is to be accepted. If the lot is accepted he gets an additional point; if not, he loses a point. Thus after the first decision his number of points is either C+1 or C-1. This procedure is repeated D times. If after D decisions the net number of points is not greater than zero, action is taken against the coder (for instance he could be retrained or completely removed from the operation).

It could be argued that in the long run every coder reaches zero. Therefore, before the next sequence of D decisions starts the initial bonus is adjusted: if the net number of points is C or more, the new initial bonus is C and if the net number is any other number in the interval [1, C-1] the new initial bonus is that number. In US Bureau of the Census applications, D has seldom exceeded 15 in order to insure against systematic deterioration after some initial good work.

The literature on the point system also includes formulas for calculating the probability of surviving D decisions, given C initial points. Furthermore, it includes tables for the number of credits expected to be accumulated by a coder still in business after D decisions and tables for the expected number of decisions given that the coder is removed on or before the Dth decision on his work. The point system could work well as a tool for process control. However, at the end of the alley we have the ghost of removal, and in many countries this ultimate solution is neither allowed nor desirable.

The point system also makes it possible to construct a quality index defined as

$$(A - R)/D$$

where A is the number of accepted lots, R is the number of rejected lots, and A + R = D. This is a kind of quick index and experience shows that it is highly correlated with the fraction of defective code numbers. Besides being used in the process control this index could be combined with production speed in order to provide a basis for incentive awards based on both quality and average production.

It is also possible to use certain non-parametric tests in the process control. For instance, runs tests, successive differences tests, and sign tests can be used as "quick" methods for controlling an operation.

If we are using a system with process control after qualification, as in the 1970 U.S. censuses, the qualification rules must be rigorous. In Minton (1970 a) two such qualification rules are presented together with probabilities of qualifying. The first rule is that s successive work lots of an individual coder have to be accepted within at most d sampling verified lots. The second rule is that at most f lots may fail the inspection within d inspected.

13.2 Limitations in the application of quality control plans

Statistical quality control in clerical operations such as coding is not always easily applied. Problems will arise in at least the following areas.

- Inspection or verification is often assumed to be perfect. If that is not the case, the OC-curve will shift and prespecified quality limits will not hold. This problem is serious if imperfect inspection methods such as dependent verification are used.

- The plans used in coding applications are most often designed as single sampling for attributes; i.e. the production code number should be classified as defective or non-defective. In a census a coder often has to code several variables for each element. These variables differ in complexity and importance and the attribute classification can be quite coarse.

These problems will now be discussed.

13.2.1 Inspection errors

It is well known that inspection or verification errors can have a serious influence on the efficiency of sampling plans for statistical quality control. The concept of inspection inefficiency was discussed as early as Lavin (1946). As we

have seen, while errors of the first type (non-defective code numbers are changed to defective ones) and of the second type (defective code numbers are not corrected) both occur in coding, errors of the second type are usually much more common than errors of the first type. Different ways of compensating for inspection errors in attribute single sampling inspection plans are presented in Minton (1972).

The effects of inspection errors on OC-curves and AOQ-curves can be rather substantial. In Minton (1972) acceptance sampling methods which compensate for erroneous inspection are presented. Modifications of the OC and AOQ curves are defined, called respectively the AOC-curve (applicable operating characteristic curve) and the AAOQ-curve (applicable average outgoing quality curve).

The method for compensating for inspection errors is to adjust the sample size while holding the acceptance number c fixed. Of course, this has an effect on the average cost per lot. The formulas given for computing these costs are reproduced in appendix III to this chapter. Let us give an example.

In the example to follow the costs are assumed to be $I_u = .3$, $C_u = .15$ and $U = 1$ (dollars, for instance), where I_u is the cost to inspect an item in a lot, C_u is the cost to correct an item in a lot and U is the average cost per lot for undetected and uncorrected defective code numbers (see appendix.) Furthermore, errors of the first type are assumed to be zero while errors of the second type are assumed to have various different values. The costs are calculated for nine different sampling plans:

(1)	N = 300	(2)	N = 1000	(3)	N = 100
	n = 30		n = 90		n = 14
	c = 1		c = 4		c = 1
(4)	N = 400	(5)	N = 100	(6)	N = 1000
	n = 29		n = 70		n = 48
	c = 5		c = 3		c = 4
(7)	N = 1000	(8)	N = 1000	(9)	N = 800
	n = 43		n = 49		n = 68
	c = 5		c = 8		c = 3

We assume that β_1 takes values .3, .5 and .7 and that β_2 takes values .1, .3 and .5 where β_1 is the probability of classifying a defective code number as non-defective when inspecting a lot and β_2 is the probability of classifying a defective code number as non-defective when rectifying a rejected lot. Thus for each plan we get nine different tables for average cost per lot. One of these tables is given by Minton and is reproduced below as Table 13.1: that for plan number 9 with $\beta_1 = .5$ and $\beta_2 = .3$.

Table 13.1 Cost per lot (T, T', T") in a single acceptance sampling plan (with errors of the second type in inspection and correction and unadjusted (n) and adjusted (n') sample sizes)

True fraction defective (P)	n = 68 c = 3 T ($\beta_1=\beta_2=0$)	n = 68 c = 3 T'($\beta_1=.5;\beta_2=.3$)	n' = 136 c = 3 T" ($\beta_1=.5;\beta_2=.3$)
0	$ 20	$ 20	$ 41
.02	45	37	64
.04	106	61	122
.06	172	95	187
.08	216	135	233
.10	239	176	260
.12	250	211	275
.14	255	241	285
.16	259	264	293
.18	262	282	300
.20	264	296	307
.22	266	308	313

As can be seen from Table 13.1 we have a break-even point at P (true fraction defective) somewhere between .14 and .16, (in fact the break-even point is P = .155). If P is less than that value the system with errors of the second type is less expensive than that with no inspection and correction errors. When adjusting the sample size the resulting cost is always higher. In Table 13.2 we list such approximate break-even points for each of the plans and their different β_1- and β_2-values.

Table 13.2 List of break-even points

β_1, β_2-combination		Plan 1	2	3	4	5	6	7	8	9
$\beta_1 = .3$	$\beta_2 = .1$	.16	.13	.22	.28	.14	.20	.24	.28	.14
$\beta_1 = .3$	$\beta_2 = .3$	.13	.12	.18	.26	.12	.18	.20	.26	.12
$\beta_1 = .3$	$\beta_2 = .5$	.12	.10	.16	.24	.10	.16	.18	.24	.10
$\beta_1 = .5$	$\beta_2 = .1$	.20	.16	.25	-	.18	.25	.29	-	.18
$\beta_1 = .5$	$\beta_2 = .3$	.16	.15	.22	.30	.15	.22	.26	.30	.16
$\beta_1 = .5$	$\beta_2 = .5$	.14	.14	.20	.28	.14	.20	.24	.29	.14
$\beta_1 = .7$	$\beta_2 = .1$	.25	.22	.28	-	.24	-	-	-	.24
$\beta_1 = .7$	$\beta_2 = .3$	.22	.20	.26	-	.21	.28	-	-	.21
$\beta_1 = .7$	$\beta_2 = .5$	.20	.19	.25	-	.19	.26	-	-	.19

We see that the break-even point value for a given plan increases as a function of the β_1-value, but at the same time decreases as a function of the β_2-value. It is also seen that the break-even point value increases with the value of c and decreases with the value of n.

Thus, knowledge of the magnitude of inspection errors is very important in designing an effective sampling plan.

13.2.2 Other drawbacks

Usually information about an element is processed for more than one variable. Of course, it may be possible to administer separate sampling plans for each of the variables simultaneously but usually it is not. In the latter case one approach is to define an element as correctly coded if all variables for that element are correctly coded, while if at least one variable is incorrectly coded the element is defined as incorrectly coded. After that an acceptance sampling plan can be used. This approach was used in the 1965 and 1970 Swedish Censuses of Population and Housing. Our experiences with that approach were not entirely favorable. Obviously, the consequences are that minor errors have the same weight as more serious errors. Thus the work of a coder who makes c+1 minor errors in a lot

is rectified while the work of another coder who makes c serious errors in another lot is accepted. This seems to be an ineffective means of sampling control. In the 1970 Swedish census, within economic activity, eleven variables were coded for each element, and since the sampling unit was 'household' it is easily realized that the probability of at least one erroneous coding within the household is rather substantial. Another approach is to select a "most important" variable and use the coding control of that as a basis for acceptance sampling. (The definition of "importance" however is a problem in itself, as discussed in part I.)

In the 1975 Swedish Census of Population a ranking of different possible errors was established in order to decreasing the amount of rectifying inspection. Thus an element was defined as erroneously coded if

(1) two or more of four specific one-digit variables were incorrectly coded, or if

(2) occupation was incorrectly coded with respect to the two first digits, or if

(3) any other variable was incorrectly coded.

This approach was possible because of the fact that, for instance, the last digit of the occupation code number is seldom used in a general presentation of census results. Statistics based on the most detailed level are merely made available on request from researchers.

Thus variables can be considered important or less important. The same is true of coding errors. Information on their importance could make possible some kind of sampling by variables plan as an alternative to the normally used attribute plans. In a variable plan each coding error would be quantified and a more solid basis for decisions on quality could be achieved.

A sampling by variables acceptance sampling plan is characterized by the fact that the quality characteristics can be measured on a continuous scale and have known distributions. The basis for decisions on quality could be the sample mean and the sample variance.

Let us make the following assumptions. We have a lot consisting of N elements. For each element information is obtained about n different variables. Each variable is either correctly or incorrectly coded. We now define the sequence of variables $X_1, X_2, \ldots, X_n$ where X_i has a two-point distribution (0, 1, p_i) and where $P(X_i=0)=p_i$ and $P(X_i=1)=1-p_i$. Thus p_i is the probability of coding variable i correctly, and $E(X_i)=1-p_i=q_i$.

Furthermore we define a sequence of weights $w_1, w_2, \ldots, w_n$ associated with the n different variables which are fixed constants proportional to the relative importance of the n variables. For each w_i the relation $0 \leq w_i \leq 1$ holds and

$$\sum_{i=1}^{n} w_i = 1.$$

If all w_i equal 1/n an attribute sampling scheme can be adopted. If not, a variables sampling scheme might be an alternative.

Now let us define the variable $v_i=w_iX_i$ which has a two-point distribution (0, w_i, p_i). All X_i are assumed independent as are all v_i.

A possible way of getting a quality measure for the coding of element j is to define

$$Z_j = \sum_{i=1}^{n} v_i = \sum_{i=1}^{n} w_iX_i$$

where j = 1, 2, ..., N.

Now

$$E(Z_j) = \sum_{i=1}^{n} w_i E(X_i) = \sum_{i=1}^{n} w_i q_i \quad \text{and}$$

$$\mathrm{Var}(Z_j) = \sum_{i=1}^{n} w_i^2 q_i p_i .$$

All $Z_1, Z_2, \ldots, Z_N$ are independent and identically distributed.

Now let us define the lot mean quality

$$y = \frac{\sum_{j=1}^{N} Z_j}{N}$$

where

$$E(y) = \frac{\sum_{j=1}^{N} E(Z_j)}{N} = \sum_{i=1}^{n} w_i q_i$$

and

$$\mathrm{Var}(y) = \frac{\sum_{j=1}^{N} \mathrm{Var}(Z_j)}{N^2} = \frac{\sum_{i=1}^{n} w_i^2 q_i p_i}{N}$$

$Z_1, Z_2, \ldots, Z_N$ are independent and identically distributed. Then, according to the central limit theorem, the variable

$$u = \frac{(y - \sum_{i=1}^{n} w_i q_i)\ \sqrt{N}}{\sqrt{\sum_{i=1}^{n} w_i^2 q_i p_i}}$$

is approximately normally distributed with expectation zero and variance equal to one, and could serve as a test variable for accepting or rejecting the lot. In order to carry out this process we need a set of weights and estimates of the different p_i. The weights should, at least in principle, be decided by

the users of statistics and the p_i-estimates can be obtained from earlier studies or from current production. Of course it may be necessary to change the weight system or the estimates during production.

13.3 A study of error importance

During the 1970 Swedish Census of Population two-thirds of the material was subject to independent verification. The first step in the verification was matching the code numbers assigned by the production coder and the verifier according to the sequential verification system adopted. The two sets of code numbers were compared for each schedule; the following cases could occur:

a) 2-0. All code numbers in the production coder set coincided with the corresponding code numbers in the verifier set.

b) 1-1. At least one code number did not coincide between the two sets.

Each 1-1 case was treated according to the specified rules. The important point was that a schedule with one coding disagreement (perhaps minor) was treated in the same way as one with, say, four coding disagreements.

The number of 1-1 cases was large in this census, and as a result the volume of rectifying inspection was large and the need for expert coders was great.

We have already mentioned that the number of 1-1 cases during the independent verification was so large that the rectification had to be abandoned for one-third of the material. Thus, if it is impossible to obtain high-quality coding the notion of quality must be softened. At least, this was the main reason for carrying out the small experiment described below. We selected 34 lots at random from the set of lots subjected to indepen-

dent verification. We noted the causes for the different 1-1 cases and we noted what would have happened if we had had a "softened" notion of disagreement, which we defined as follows:

i) The last digit of the three-digit occupation code and the last two digits of the four-digit industry code were skipped when comparing the code numbers assigned; i.e., only the two first digits were considered in the matching;

ii) Coding disagreements in at most one of the remaining variables (i.e. other than occupation and industry) would not be considered a 1-1 case.

Let us pick two of the 34 lots and see what happened.

Lot 9

The number of schedules sampled was 27. The number of 1-1 schedules was 11.

		Kind of initial disagreement
Schedule	1	More than two "other" disagreements
	2	More than two "other" disagreements
	3	Industry, second digit
	4	One "other" disagreement
	5	One "other" disagreement
	6	One "other" disagreement
	7	More than two "other" disagreements
	8	One "other" disagreement
	9	Occupation, last digit
	10	Occupation, first digit
	11	One "other" disagreement

With our softened notion of quality, schedules 4, 5, 6, 8, 9 and 11 would not have been considered 1-1 cases; thus the "revised" number of 1-1 cases is 5 (schedules 1, 2, 3, 7 and 10).

Lot 32

The number of schedules sampled was 22. The number of 1-1 schedules was 9.

		Kind of initial disagreement
Schedule	1	More than two "other" disagreements
	2	Two "other" disagreements
	3	One "other" disagreement
	4	Two "other" disagreements
	5	One "other" disagreement and one reject in OCR-scanning
	6	One "other" disagreement
	7	One "other" disagreement
	8	Industry, first digit
	9	One "other" disagreement

With our softened notion of disagreement, schedules 3, 6, 7 and 9 would not have been considered 1-1 cases and thus the revised number of such cases is 5. (Disagreements due to the OCR equipment were considered coding disagreements until they were properly identified.)

In this study, which is described in Lyberg (1974 c), the total number of schedules sampled was 798, and the number of 1-1 cases, according to the usual definition, was 329. Thus an estimate (although with large variance) of the proportion of schedules demanding further work is an amazing 41 %. The revised number of 1-1 cases is 188 and the estimated proportion decreases to 23.5 %. This considerable reduction is due to the special distribution of disagreements. On 176 of the 329 1-1 schedules there was only one disagreement, on 68 there were exactly two, and on 56 there were three or more. The remaining 29 were so-called "singles" - rejects in the optical scanning. This study, together with some additional information, led to the alternative error definition used in the 1975 Swedish census which was described in 13.2.2 above.

APPENDIX III

FORMULAS FOR COMPUTING AVERAGE COST PER LOT

Operation	Inspection and correction, error-free	Errors of the second type in inspection and correction	
		Unadjusted sample	Adjusted sample
Inspection	$I=[n+(N-n)\ (1-L_p)]I_u$	$I'=[n + (N-n)\ (1-L_{p'})]I_u$	$I''=[n' + (N-n')\ (1-L_{p''})]I_u$
Correction	$C=[n+(N-n)\ (1-L_p)]PC_u$	$C'=[n(1-\beta_1)P+(N-n)\ (1-L_{p'})(1-\beta_2)P]C_u$	$C''=[n'(1-\beta_1)P+(N-$ $(1-L_{p''})(1-\beta_2)P]$
Non-detection and correction	$U=[(N-n)L_p]PU_u$	$U'=[(N-n)L_{p'}P+n\beta_1P +(N-n)(1-L_{p'})\ \beta_2P]U_u$	$U''=[(N-n')L_{p''}P+ +n\beta_1P+(N-n)\ (1-L_{p''})\beta_2P]U_u$
Total	$T=I+C+U$	$T'=I'+C'+U'$	$T''=I''+C''+U''$

The definitions of the symbols are:

P = true fraction defective

N = number of items in the lot

β_1 = probability of classifying a defective code number as non-defective when inspecting a lot

β_2 = probability of classifying a defective code number as non-defective when rectifying a rejected lot

n = lot sample size

n' = sample size, adjusted for inspection error β_1 ($n' = 1/(1 - \beta_1)$)

L_p = probability of accepting a lot (size n) without inspection error

$L_{p'}$ = probability of accepting a lot (size n) with inspection error

$L_{p''}$ = probability of accepting a lot (size n') with inspection error ($L_{p''} \approx L_p$)

I_u = cost to inspect an item in a lot

C_u = cost to correct an item in a lot

U_u = cost attributed to a defective item remaining in a lot after sample inspection and correction

I = average inspection cost per lot (n, N, L_p)

C = average correction cost per lot (n, N, L_p)

U = average cost per lot for undetected and uncorrected defective code numbers (n, N, L_p)

T = average total cost per lot (n, N, L_p)

I' = average inspection cost per lot (n, N, β_1, β_2, $L_{p'}$)

C' = average correction cost per lot (n, N, β_1, β_2, $L_{p'}$)

U' = average cost per lot for undetected and uncorrected defective code numbers (n, N, β_1, β_2, $L_{p'}$)

T' = average total cost per lot (n, N, β_1, β_2, $L_{p'}$)

I" = average inspection cost per lot (n', N, β_1, β_2, $L_{p''}$)

C'' = average correction cost per lot $(n', N, \beta_1, \beta_2, L_{p''})$

U'' = average cost per lot for undetected and uncorrected defective code numbers $(n', N, \beta_1, \beta_2, L_{p''})$

T'' = average cost per lot $(n', N, \beta_1, \beta_2, L_{p''})$

C SOME ADDITIONAL TOPICS

Some topics remain to be discussed in describing manual coding. In the three subsequent chapters some material in the following areas will be presented.

First, a more detailed illustration of errors in coding is given. Most of this illustration has to do with evaluation of coding, which can have an impact on preventive measures taken in future coding operations. Then some ideas concerning the use of prior information when designing control operations are put forward. Finally, the method of decentralized coding is described. Although this method is fairly common survey practice, the lack of performance studies is striking.

14 Errors in coding: detailed discussion

In this chapter errors in coding will be dealt with in more detail, most of the illustrations will be taken from Swedish census studies.

14.1 Estimating error rates

Estimating coding error rates implies the existence of a correct instrument which generates true code numbers. Although, in coding as we have seen, it is difficult to develop a correct instrument, there do exist a couple of operational routines designed to provide, if not true code numbers, at least preferred code numbers. It is assumed that these preferred code numbers are better than those obtained by the fallible instrument (for instance, the production or regular coding), and therefore can be used for evaluation purposes.These operational routines are:

i) A superexpert, perhaps the designer of the code, decides the true code numbers. This approach can be unrealistic in at least two respects. First, there are no real experts, in the sense that no one has unit probability of correct coding. Second, the work load in an evaluation study for the most expert person available might be prohibitively large.

ii) A pool of coders assigns the set of evaluation code numbers by means of some decision rules as discussed in chapter 12. This procedure can be spurious since even good and experienced census coders produce considerable within - and between - coder variations.

By using such procedures we obtain the operationally true coding. By comparing the operationally true coding with the regular coding, different measures of error rates can be obtained. The degree of agreement between regular code numbers (R) and evaluation code numbers (E) (operationally true coding) for a specific variable can be illustrated in a table of the following type.

R \ E	1	2 . . .	i	K	Σ
1	n_{11}	n_{12}	n_{1i}	n_{1K}	$n_{1.}$
2	n_{21}	n_{22}	n_{2i}	n_{2K}	$n_{2.}$
.					
.					
.					
i	n_{i1}	n_{i2}	n_{ii}	n_{iK}	$n_{i.}$
.					
.					
.					
K	n_{K1}	n_{K2}	n_{Ki}	n_{KK}	$n_{K.}$
Σ	$n_{.1}$	$n_{.2}$	$n_{.i}$	$n_{.K}$	$n_{..}$

A simple measure of the error rate is

$$Z_{RE} = 1 - \frac{\sum_{i=1}^{K} n_{ii}}{n_{..}}$$

where n_{ii} is the number of elements coded i by both R and E and $n_{..}$ is the total number of elements coded.

This measure is elaborated upon somewhat in Crittenden and Hill (1971).

In the evaluation of the 1970 Swedish and US censuses, procedure ii) was used to provide preferred code numbers. In US Bureau of the Census (1972 b) estimates of possible bias between the regular code major group and the evaluation code major group for occupation and industry are given. For almost every major group the 95 % confidence interval for the bias included zero.

Measuring error rates should not be restricted to production coders only. We have already seen that verification is seldom perfect. By means of procedure ii) it is also possible to evaluate the performance of verifiers. Some results were given in chapter 11. An alternative procedure for the estimation of verifier error rates is error planting. With this procedure known defective code numbers are deliberately planted in the assembly line; i.e., the inspector or verifier is presented with items some of which have erroneous code numbers. The point is that the administrator knows which items are defective. After the inspection the verification is evaluated. Suppose that f defective items are planted, and the verifier detects $f' \leq f$ of these: then the verification error rate is f'/f. As discussed above, this type of estimate is valid only for the types of errors that are planted. Naturally, in practice one hopes that the planted errors are representative of the set of real errors.

An alternative test of verifiers is the following (Berg (1974)). A large set (N) of correctly coded items is available. Into this set F erroneously coded items are planted at random. From N a random sample of n is drawn for verification purposes and $n/N=p$. Each item is associated with one code number.

This sample is verified by r different verifiers, and it is observed that verifier number i identifies x_i erroneous code numbers, i = 1, 2,..., r. For illustration purposes x_i can be regarded as an observation from a Poisson distribution with $\lambda = n\ F/N = p\ F$ (although a hypergeometric distribution would be more appropriate). The basic assumption is that $E(x_i) = p\ F$.

As a consequence verifier number i produces an estimate of F

$$\hat{F} = x_i/p \qquad i = 1, 2,..., r$$

with $\mathrm{Var}(\hat{F}_i) = \mathrm{Var}(\frac{N}{n}\ x_i) \approx \frac{N^2}{n} \cdot \lambda = F/p$

Now alternative hypotheses about the verifiers can be tested. For instance,

$$H_o: E(\hat{F}_i) = F$$

Observe that with this approach it is not necessary to know the number of erroneous code numbers given to verifier i. It is a one-time effort after which a large number of verifiers can be tested.

Let us give a numerical illustration for verifier i.

$N = 10000$
$F = 1000$
$n = 200$
$p = .02$
$x_i = 14$
$\hat{F}_i = 14/\ .02 = 700$

A simple z test gives

$$U_{obs} = \frac{700-1000}{\sqrt{50 \cdot 1000}} \approx -1.34$$

This value is not significant at, for instance, the 5 % level and we cannot reject H_o.

Now suppose that verifier j detects 7 erroneous code numbers, i.e. $x_j = 7$. With the same values of N, F, n, and p as before we get

$$U_{obs} = \frac{350-1000}{\sqrt{50 \cdot 1000}} \approx -2.91$$

which is significant at the 5 % level. Thus H_o is rejected and it is a reasonable conclusion that verifier j tends to let erroneous code numbers remain unchanged. His results yield an F-value below 1000.

The main drawback to this approach is that it may be difficult to produce the large set N of correctly coded items. A more practical approach would be the following. By means of some evaluation method the error rate in the coded material is estimated as $\hat{F}_e$. This estimate can be compared to the different $\hat{F}_i$ and H_o: $E(\hat{F}_i) = \hat{F}_e$ can be tested. Negative significant values of the test indicate that the verifier tends to let errors remain in the material. Positive significant values indicate that the verifier has a tendency to insert errors.

Error planting can also be compared to a capture-recapture situation. In a large set of coded items N, an unknown number is erroneous. The question is, what is this number?

Suppose we have two verifiers working independently. Verifier number one identifies x_i erroneous code numbers out of n_1. The n_1 items are put back and verifier number two identifies x_2 erroneous code numbers out of a new sample n_2. Of these x_2, x_{21} have also been detected by the first verifier.

Now $x_1 + x_2 - x_{21}$ gives information about the unknown number. According to Polya (1976), the number of erroneous code numbers that remain undetected is $(x_1 - x_{21})(x_2 - x_{21})/x_{21}$. It is possible to have r verifiers and carry out the experiment sequentially, continuing until a prespecified number of erroneous code numbers have been identified.

The literature on inspector performance in industrial applications is relatively meagre; see however Weaver (1975), where several different measures of error rates are discussed. With modifications some of them could be useful in clerical situations as well.

Another problem with error rate calculation is the fact that coding performance differs between different coding teams. In large-scale coding, in a census for instance, the coding is carried out by different crews or teams and each crew has its own leader or supervisor. During the 1965 Swedish census the coding took place on two different floors in the same building. It was the general feeling among the census administrators that different coding patterns had developed on the two floors, although this feeling was not supported by any data. The same feeling is expressed in Krause (1971) where problems with the 1970 US census coding are discussed. For example, the Place of Work supervisors in one bay were giving their coders instructions opposite to what supervisors in another bay were telling their coders, resulting in different "true" code number assignments for the same descriptions. In the 1970 Swedish census this problem was enhanced in that the coding took place in two different cities 150 miles apart, and moreover within each city, it took place in different buildings. This was radically changed for the 1975 Swedish census: all coding took place in one city, in one building and on one floor. This change could be a major reason for the great reduction in the 1975 coding error rate. During the 1970 Swedish census a special study (Lyberg (1972 a)) was carried out to see whether

coding quality differed between the two cities. For a relatively small sample the number of schedules containing at least one coding error was counted. The difference between the two cities was an estimated 3.4 percent, not a significant difference.

The problem with different buildings and different floors may be viewed as a problem of dependence. The following recent study (Lyberg et al. (1978)) might be another instance of this phenomenon.

The survey under study is "survey of pupils graduated from the 9-year compulsory comprehensive school" in which the respondents were asked about their present occupation and planned occupation. The coding of these variables was carried out by Statistics Sweden. We were interested in evaluating the quality of the coding and a minor experiment was conducted.

A random subsample of 1 006 individuals was drawn from the original sample. All these have a "planned occupation". Some of them, but not all (not those unemployed, still studying, etc.), have a "present occupation".

In order to obtain "true" code numbers, we first asked two coders to code the material independently. The first coder belonged to the group which usually codes the survey, and the other coder belonged to another coding team (one which used the same code in coding occupations in other surveys). The code numbers were matched and identical code numbers were accepted as true ones. The non-identical code numbers were recoded by an "expert team" consisting of two coders from the pupil survey coding team. These coders had access to both the verbal descriptions and the primary code numbers. They were asked to note the causes of the discrepancies. It appeared that in most cases coders had chosen the occupation code number assigned by the coder from their own group (they knew which number that

was). Therefore, we asked another "expert team" consisting of two census coders to recode the material. This team also had access to the two primary code numbers, but not to the code numbers assigned by the first expert team. Table 14.1 shows the results of the experiments; in that table

P1 = code number assigned by the primary coder from the ordinary coding team;

P2 = code number assigned by the other primary coder;

E1 = code number assigned by the "expert team" from the ordinary coding team;

E2 = code number assigned by the "expert team" consisting of census coders.

Table 14.1 Results of coding experiments (occupation on the 3-digit level).

	Planned occupation		Present occupation	
	Number	% of coded cases	Number	% of coded cases
Total number of cases	946	100.0	724	100.0
P1 = P2	837	88.5	635	87.7
P1 ≠ P2	109	11.5	89	12.3
P1 ≠ P2, E1 = P1	71	65.1	54	60.7
" E1 = P2	29	26.6	30	33.7
" E1 ≠ P1, P2	9	8.3	5	5.6
P1 ≠ P2, E2 = P1	59	54.1	44	49.4
" E2 = P2	33	30.3	33	37.1
" E2 ≠ P1, P2	17	15.6	12	13.5
P1 ≠ P2, E1 = E2	62	56.9	56	62.9
" E1 ≠ E2	47	43.1	33	37.1

This table shows that the expert teams had a tendency to choose one of the previously assigned code numbers, but not always the same one. This is consistent with the results of other studies which have shown that dependent verification is inferior to independent verification and that there exists variability between "expert" coders.

14.2 Measuring variability

Variability can be illustrated by comparing results from repeated measurements by the same instrument. In variability studies we do not explicitly relate variability to true coding and therefore usually have no prospects of detecting or estimating coding bias. Variability in coding can be analyzed by replication studies in at least two dimensions, trials and agents.

Replication studies have been carried out in connection with the evaluation of the 1965 and 1970 Swedish Censuses of Population. The variability is measured by a simple proportion

$$P = m/n$$

where n is the total number of items coded for the specific variable and m is the number of items differently coded when comparing one coding with its replication. In chapter 5 some results from replication studies were presented. There it was argued that P-values for different variables are not immediately comparable, since P has a tendency to increase with the number of possible categories. Thus a seemingly low value of P for an easy variable with only a few possible categories might indicate greater coding problems than for a difficult variable with hundreds of categories, where we may obtain high P-values.

In Bennett et al. (1954) an index which corrects this distortion is used:

$$S = \frac{K}{K-1} (1 - P - \frac{1}{K})$$

where K is the number of available categories. The index is more or less built on the assumption that all the categories are equally probable. However, this is not a realistic assumption in most situations: on the contrary, clustering often occurs and the resulting distribution is far from uniform.

In Scott (1955) an alternative index is discussed which corrects for the number of available categories:

$$\pi = \frac{1-P-P_x}{1-P_x}$$

where P_x is the percent agreement to be expected by chance. P_x depends on the number of categories and on the frequency with which they are used by the coders. P_x should be calculated on the basis of a "true" distribution but since this distribution does not exist it must be calculated on the basis of an approximation. One approximation is the distribution for each coder. In the evaluation of the coding of the 1970 Swedish Census of Population five "expert" coders were used in a between-coder variability study. Each coder generated a distribution of occupation for the individuals in the sample consisting of 914 individuals. The distributions for the ten major occupation categories are found in Table 14.2 below.

Table 14.2 Five distributions of major group occupation

	Percent of the individuals in the sample				
Code number	Coder 1	Coder 2	Coder 3	Coder 4	Coder 5
0	18.9	18.4	18.2	18.4	18.4
1	.8	.6	.8	.8	.6
2	7.5	7.9	7.4	7.6	7.7
3	8.2	8.7	8.2	8.3	8.3
4	.9	1.1	1.1	.9	1.1
5	0	0	0	0	0
6	6.9	6.6	7.0	7.4	6.8
7	23.0	23.1	24.2	23.7	23.0
8	11.7	12.1	11.5	11.3	12.4
9	22.2	21.5	21.6	21.5	21.7

The between-coder variability is probably extensive. P' (as defined in 5.2.1.3) was calculated as about 10-14 percent in earlier studies, but this table indicates rather small net errors. In general, P_x equals the sum of the probabilities of agreement on each of the categories taken individually. Since the five distributions are very similar it seems that P_x could be calculated on any of these distributions or combinations of them. In fact the different P_x lie in a narrow interval (see Table 14.3).

Table 14.3 P_x-values for five expert coders.

Coder	P_x
1	.1688
2	.1664
3	.1688
4	.1673
5	.1668

If, for example, two coders agree in 95 % (1-P = .95) of the cases (P' above was calculated for five independent coders), then the highest and the lowest P_x-values are

$$\pi|(P_x = .1664) = .9400$$

and

$$\pi|(P_x = .1688) = .9398$$

Thus $\pi|P_x$ is approximately the same in the two cases; for small variations in the basic distributions this is a general fact. The variance is

$$\sigma_\pi^2 = \left(\frac{1}{1-P_x}\right)^2 \frac{(1-P)P}{n-1}$$

The significance of the difference between two π's can be calculated by the test variable

$$\frac{\pi_1 - \pi_2}{\sqrt{\left(\frac{1}{1-P_{x1}}\right)^2 \frac{(1-P_1)P_1}{n_1-1} + \left(\frac{1}{1-P_{x2}}\right)^2 \frac{(1-P_2)P_2}{n_2-1}}}$$

A similar reliability index is discussed in Crittenden and Hill (1971). This index is

$$R = \frac{D_{max} - D_0}{D_{max}}$$

where D_0 is the number of observed coding differences when code numbers are compared pair-wise and D_{max} is the maximum number of differences that occur.

14.3 The error distribution

In large-scale multipurpose surveys some variables are more important than others either because more users are involved or because information about some variables is more important to solve specific problems. Often the one-digit variables are

considered less important than multi-digit variables. In practice the former are relatively easy to code and as a consequence the error rate is rather low.

When we are dealing with variables such as occupation and industry we must use hundreds of categories. The codes must therefore be multi-digit and they are built upon the principle of Chinese boxes; i.e., the first digit in the code associates the element with a major group, the second digit in the code associates the element with a subgroup, and so on. Statistical tables often present only the major group data. Data for subgroups and minor groups, i.e. those represented by the last digits in the code, are available on request for special purposes. The conclusion must be that errors in major group coding are more serious than errors in the last digits. In addition, of course an error in a major group automatically results in errors in subgroups and decimals. For most purposes errors in, say, the third or fourth digit (or sometimes the fifth or sixth) are of little vital importance. A self-evident question arises: are most of the errors in occupation coding, for instance, concentrated in the last digits? If so, an estimated total error rate of, say, 12 percent is very misleading.

In the evaluation of the 1965 Swedish census the error distribution for the three-digit variable industry was studied. The three digits represent a major group, a subgroup, and a minor group within the subgroup (decimal). The error distribution is displayed in Tables 14.4 and 14.5. The results are given separately for the four different subsamples and for the regular coder and the verifier.

Table 14.4 Error distribution for industry coding in the 1965 Swedish census, regular coder.

Sample	Major group	Subgroup	Decimal	Total number of errors
1	41	17	22	80
2	22	8	7	37
3	37	17	20	74
4	28	8	16	52
Unweighted total	128	50	65	243

Table 14.5 Error distribution for industry coding in the 1965 Swedish census, verifier.

Sample	Major group	Subgroup	Decimal	Total number of errors
1	28	9	16	53
2	18	7	8	33
3	27	10	15	52
4	23	4	15	42
Unweighted total	96	30	54	180

This experiment was repeated during the evaluation of the 1970 Swedish census coding. This time occupation (a three-digit code) was also investigated, and by this time the industry variable had become four-digit, the first two digits representing the major group and the last two the subgroup. The results are given in Tables 14.6 and 14.7 below. The results here are given for different subsamples after verification.

Table 14.6 Error distribution for occupation coding in the 1970 Swedish census, after verification.

Sample	Major group	Subgroup	Decimal	Total number of errors
1	67	36	56	159
2	84	39	52	175
3	68	13	35	116
Unweighted total	219	88	143	450

Table 14.7 Error distribution for industry coding in the 1970 Swedish census, after verification.

Sample	Major group	Subgroup	Total number of errors
1	77	22	99
2	88	35	123
3	56	18	74
Unweighted total	221	75	296

This procedure was repeated in the 1975 Swedish census. The evaluation study gave the results, shown in Tables 14.8 and 14.9.

Table 14.8 Error distribution for occupation coding in the 1975 Swedish census, after verification.

Major group	Subgroup	Decimal	Total number of errors
138	68	116	322

Table 14.9 Error distribution for industry coding in the 1975 Swedish census, after verification.

Major group	Subgroup	Total number of errors
83	61	144

The tables show a pattern which is quite clear: serious errors are most frequent. This might seem remarkable at first glance since intuitively it does not appear to be so difficult to choose between, say, ten major occupation groups. One would expect that this would be carried out correctly and that then some difficulties might perhaps appear in the details of the code. On the contrary, things seem to work the other way: provided the major group is identified correctly it is relatively easy to code the details. This indicates that we have problems with the code dictionary and that the dividing-lines between major categories may be diffuse. This problem will be discussed in the next chapter.

It is well known that the errors are not evenly or proportionally distributed among the different categories. Certain categories give rise to more erroneous coding than others. This phenomenon can be measured by means of an index of consistency. One example of such an index can be found in Fasteau et al. (1962) and Hansen et al. (1964). The index answers the question, "given a specific code number, how consistently was it independently applied by a number of different coders looking at the same verbal description?" For three coders the index equals

$$I = \frac{3\,N_{AAA} + N_{AAB}}{3\,N_{AAA} + 2\,N_{AAB} + N_{ABB} + N_{ABC}}$$

where N_{AAA} is the number of items for which all three coders agreed upon the specific code number,

N_{AAB} is the number of items for which two coders agreed upon the specific code number while one disagreed,

N_{ABC} is the number of items for which all three coders disagreed but one of them assigned the specific code number,

N_{ABB} is the number of items for which only one coder assigned the specific code number while the other two agreed upon some other code number.

The index helps to identify those code numbers which have low reliability because of confusion on the part of the coders. In a study discussed in Fasteau et al. (1962) it was found that every industry code number for "wholesale trade" had a value under a pre-specified critical limit I = .80. This kind of information can be used to design a better code and better coding instructions. The frequencies N_{AAB}, N_{ABB} and N_{ABC} give information about associate code numbers, i.e. code numbers which have a propensity to be interchanged.

The derivation of the index can be found in Fasteau et al. (1962).

Table 14.10, excerpted from the same reference, is illuminating. It presents the percentage of industry and occupation code numbers, respectively, by index of consistency, for the coding operation of the 1960 US Censuses of Population and Housing.

Table 14.10 Distribution of industry and occupation code numbers with respect to index of consistency, the 1960 US Census of Population.

Index of consistency	Per cent of code numbers	
	Industry	Occupation
.900- .999	39.3	48.0
.800- .899	37.3	31.4
.700- .799	15.3	13.9
.600- .699	4.1	3.7
.500- .599	1.3	1.3
<.500	2.7	1.7
Total	100.0	100.0

As pointed out in Swensson (1969), there are certain interpretation difficulties associated with the use of an index of consistency: such an index depends in general not only on the reliability of the coding process but also on the proportion of population elements belonging to the category under study.

During the last three Swedish censuses we have studied the propensity to differ for specific code numbers. No special pattern was observed for industry in 1965.

However, in Lyberg (1974 b) and in Andersson (1977) the evaluation major group code numbers for occupation were compared with the outgoing production code numbers in the 1970 and 1975 censuses. The results are presented in Tables 14.11 and 14.12.

Table 14.11 Distribution of erroneous code numbers for occupation, major group, 1970 census.

"True" code number \ Production code number	0	1	2	3	4	5	6	7	8	9	Total
0	576	5	14	4	3	0	3	17	5	5	632
1	2	54	4	1	0	1	0	1	1	0	64
2	11	0	302	3	0	0	6	2	0	2	326
3	9	3	4	265	0	2	1	3	2	1	290
4	3	0	0	0	208	0	0	0	1	3	215
5	0	0	0	0	1	4	0	0	1	0	6
6	1	1	2	1	0	0	205	1	4	1	216
7	12	1	2	2	1	0	2	768	9	2	799
8	4	2	0	2	2	0	0	11	334	1	356
9	22	0	0	5	0	0	2	1	1	367	398
Total	640	66	328	283	215	7	219	804	358	382	3302

Table 14.12 Distribution of erroneous code numbers for occupation, major group, the 1975 census.

"True" code number \ Production code number	0	1	2	3	4	5	6	7	8	9	Total
0	862	4	6	2	0	0	0	2	2	4	882
1	5	74	4	3	0	0	0	0	0	0	86
2	5	5	472	8	1	0	3	1	0	1	496
3	8	1	4	339	0	0	2	4	0	0	358
4	0	0	0	0	339	0	0	0	1	0	340
5	0	0	0	0	0	8	0	0	0	0	8
6	1	0	4	0	0	0	226	1	2	3	237
7	4	0	1	0	0	0	1	801	5	0	812
8	2	0	1	3	1	0	2	16	310	1	336
9	5	1	0	2	2	0	1	2	1	537	551
Total	892	85	492	357	343	8	235	827	321	546	4106

As can be seen from the tables the marginal distributions are rather similar; this illustrates the fact that gross errors cancel out to some extent in statistical tables.

In an evaluation study (Olofsson (1976)) of the industry coding of the Swedish Labor Force Survey the average gross error was estimated to 5.7 %. Naturally for most industry code numbers the net error is considerably lower than the gross error: one extreme example is "retail trade" which has an estimated gross error rate reaching 7.6 % but a zero net error. The error variation between code numbers is rather small. With a few exceptions the gross error rate varies between 1.7 and 7.9 %. The net error varies generally between zero and 3.6 %. However, a few categories are heavily over- or underestimated. The worst examples are "forestry" and "housework" with estimated net errors reaching 12 and 30 % respectively. If we consider the major groups only, the average gross error is estimated to 2.2 %. The net errors

vary between .2 and 1.3 % and the 95 % confidence intervals for the net errors all include zero.

In cases when no evaluation code numbers exist it is still possible to detect at least whether coder disagreements have systematic or random causes. In Funkhouser and Parker (1968) such a method is described.

15 Use of prior information when designing schemes for manual coding operation

As evidenced in the realm of industrial quality control, "prior information" may be put to excellent use in designing schemes for quality control. One may therefore argue that there may be various ways of using prior information in the design of schemes for control of coding. It is the purpose of this chapter to discuss this subject for the case of manual coding.

15.1 The coding environment

In Krause (1971) some serious problems with the coding in the 1970 US census are listed. Among other things, the computer programs were not adequatly tested prior to the start of the operation. This eventually led to the abandonment of parts of the independent verification system. Another problem was that the quality control program was not fully understood by the clerical staff. This latter problem was also present in the 1970 Swedish census. In Sweden we also had the problems associated with coding in different cities. Furthermore, the financial implications of the suggested independent verification systems had not been studied in detail. Eventually this resulted in the abandonment of the rectifying inspection for a huge part of the coded material. Finally we had problems with the system for optical character recognition of assigned code numbers. While we knew of course that the code numbers could be optically recognized, we had no experience in large-scale use of such systems, and thus, when for instance optical forms were destroyed or simply disappeared we were trapped. We had to solve such problems as they appeared on the scene.

During the 1965 Swedish census we let the verifiers do the sampling of items to be verified. This approach might seem stupid in retrospect but at the time it seemed practical. Anyway, as a byproduct of the coding evaluation we found that the ver-

ifiers did not always pick every tenth schedule (the sampling fraction decided). Sometimes they took a schedule nearby. In some cases we could note that these nearby schedules appeared easier to code than those which should have been sampled. In a special study 17 lots of verified schedules were checked and we found that in 9 of these the verifier had made at least one mistake (deliberate or not) during the sampling process. One example is the following lot.

Lot 14/80/258

Sampling plan:

Schedules number	5	15 ...	95	105 ...	145	155 ...	235	245 ...	275

Actual selections:

Schedules number	5	15 ...	93	103 ...	146	156 ...	237	247 ...	277

Thus in this lot with an intended systematic sample of every tenth schedule with a random starting point at schedule number 5, the verifier made three mistakes. These findings support the suggestion that in a pure manual system the sampler and the verifier should not be the same person.

15.2 Dictionaries and other coding means

Evaluation studies can give hints toward certain improvements in code construction and the design of other coding means as discussed in earlier chapters.

15.3 The verification system

In chapter 11 it was pointed out that dependent verification is not always better than no verification at all in terms of outgoing quality. For dependent verification to improve quality, the relation

$$Q\beta + (1-Q)\alpha < Q$$

must hold, where Q is the proportion of erroneous code numbers in the population previous to verification, α is the probability that a correct code number is changed by a verifier and β is the probability that an erroneous code number is not correctly changed by a verifier. Obviously this relation consists of unknown quantities which have to be estimated in order to obtain a basis for a sound decision. Again, evaluation and pilot studies are proper means to obtain such estimates.

In chapter 13 we discussed the efficiency of sampling plans with the presence of inspection errors. Since inspection errors make the OC-curve shift, it is necessary to have some knowledge about, for instance, Π (the proportion of erroneously coded items corrected by the dependent verifier) in order to obtain the desired quality guarantee for the sampling plan.

In Sirken (1973) the dependent verification of the coding of the Current Mortality Sample is discussed. There it is shown that a dependent system on a 10 % sampling basis gives almost the same outgoing quality as a 100 % system provided a production error rate of 2 % and that the cost is much less. It is also argued that dependent verification should be preferred to independent verification as long as the coding quality prior to verification is very high. (Again, we are reminded of the importance of having good input.)

Schemes for independent verification have been discussed in chapter 12. The schemes are associated with some basic measures and prior information about these if the schemes are to be effectively applied. For instance, there are three different categories of coders in the pool. It is assumed that the probabilities p_{ij}, q_{ij} and t_{ij} exist and implicitly that these probabilities differ. We must have estimates of these probabilities to be able to calculate expected costs and quality given

a specific scheme. Of course such estimates will not be needed if it is only a question of comparing schemes in order to select the best one. But if the purpose is to determine just how good a scheme is, then again these estimates are needed.

The estimates could be obtained by using a master set of the type described in chapter 10. The estimation process is very simple. The result of the coding operation is compared to the true master set coding. Then it is also possible to discover if the assumption that all coders within a specific category have the same, or almost the same, coding ability holds. Furthermore, the implicit relation

$$p_{ii} < q_{ii} < t_{ii}$$

can be verified.

Costs and speed can also be estimated, but one must remember when using the master set coding for estimation purposes including probabilities, that the set must be large enough and must have essentially the same structure as the production material in order to yield reasonable accuracy.

15.4 Acceptance sampling

The efficient choice of an acceptance sampling plan assumes that the error rate p before verification is known in advance. If p is underestimated much rectifying inspection is needed. If p is overestimated the resulting sample size is unnecessarily large. The estimation of p is a difficult task in a coding operation. Evaluations of the process show that p varies a great deal over time, between coders, and between different lots (as in the coding of industry in urban areas compared to rural areas).

Again the master set could be used; while it is doubtful whether it is an efficient method of estimating the process average p, it ought to be better than the alternative of guessing wildly. It is common practice to accept an overestimation of p together with its economic consequences. This was the procedure in the 1965 Swedish Censuses of Population and Housing. In the 1970 census we underestimated p and ended up with much rectifying inspection, which also was very unfortunate from an economic point of view.

15.5 Bayesian methods

As we have discussed above, statistical quality control originally designed for industrial purposes is a possible instrument for control of a coding operation. Prior knowledge of the proportion of defective code numbers is necessary in order to obtain an optimal plan. Usually the proportion of defective code numbers is not considered to be a stochastic variable; instead, one usually picks a plan with too large n.

In Hald (1960), (1963) and Guild (1976), for instance, it is argued that sampling plans can be designed more effectively by using a prior distribution for the process average p and by considering this quantity to be a stochastic variable. Traditional quality control systems in the spirit of Dodge-Romig deal with rather vague assumptions. Their prior distribution consists of a process average p and one part illuminating lot tolerance with weights unspecified. Hald claims that an effective sampling plan must be based on a prior distribution for p together with a picture of the economic consequences of rejects and acceptances. He uses the compound hypergeometric distribution which has the hypergeometric, the binomial, the Polya and the rectangular distributions as special cases. Guild, on the other hand, uses a gamma prior distribution. Of course the Bayesian approach depends on how well the prior distribution chosen

fits the process. However, a body of information can be built up continuously when we do lot sampling and it would be valuable if these "historical" data could be used in the control process in order to reduce the sample size. In fact, statistical quality control seems to be one of the very few areas where both Bayesians and many non-Bayesians agree that the assessment of a prior distribution might be all right.

In Barnett (1973) it is discussed how prior information about the proportion of defectives could be used. Suppose the proportion of defectives (erroneously coded elements) before verification in batch (k+1) is Θ_{k+1}. A simple point estimate of Θ_{k+1} could be obtained by sampling elements from batch (k+1). Then an estimate of Θ_{k+1} is $\hat{\Theta}_{k+1} = r/n$, where r is the number of defective elements and n is the sample size. In many situations we may have specific information about the quality in the earlier batches 1, 2, ..., k: for instance, it is possible that these k earlier batches were totally verified, while from batch (k+1) and on we will have sampling verification. Thus it is possible that we have a set of values $\Theta_1, \Theta_2, \ldots, \Theta_k$ which is known. The distribution of these Θ-values can now be considered a prior distribution where Θ_{k+1} is an observation from that distribution. The set $\{\Theta_i\}$ could be represented by, for instance, a histogram for which a continuous function could be fitted and this function could serve as a prior distribution for batch quality.

Often the proportion of defectives in a batch can be represented by beta distributions. In Barnett (1973) and Hendricks (1964) examples of such situations are given. Bayesian analysis in acceptance sampling situations is further discussed in Hald (1968) and Weiler (1965).

A similar approach is the following. Suppose we want to estimate the coding error percentage in a certain production. A sample of n elements is drawn. An effective verification shows that

m elements are erroneously coded. According to standard techniques the percentage is estimated by $\hat{p}=m/n$. But suppose we have information from an earlier coding operation of the same kind and that we know the error rates provided a certain number of coded elements. Thus the following table is known:

P_j (error percentage)	$P_o(p_j)$ (relative frequency)
P_1	$P_o(p_1)$
P_2	$P_o(p_2)$
.	.
.	.
P_l	$P_o(p_l)$
	$\sum_{j=1}^{l} P_o(p_j)=1$

For practical purposes we must reduce the set of p_j-values in order to get, for instance, integer values. When constructing the table we could adjust observed p_j-values to the nearest integer. Then we get a table which is a discrete prior distribution for p.

For every p_j-value, $P(x=m|p_j)$ is calculated, where x is the number of erroneously coded elements in the sample. Assuming independence leaves us with

$$P(x=m|p_j) = \binom{n}{m} p_j^{\,m} (1-p_j)^{n-m}$$

According to Bayes' theorem the following expression is obtained:

$$P'(p_j) = P(p_j|x=m) = \frac{P(x=m|p_j)P_o(p_j)}{\sum_{j=1}^{l} P(x=m|p_j)\, P_o(p_j)}$$

An estimate of the coding error rate is

$$\hat{p}' = \max_j \{P'(p_j)\}$$

16 Decentralized coding

Centralized coding is a very expensive operation. In the 1970 Swedish population census, for example, the cost exceeded 5 000 000 kronor (≈$ 1 200 000). It is therefore natural to look for alternatives which may significantly reduce the cost. We will consider here the use of decentralized coding. More specifically, we will consider three types of such schemes:

i) The respondents themselves are requested to tick a box or (more interesting from our viewpoint) to code by means of a dictionary

ii) The interviewers are instructed to code certain variables by means of a dictionary

iii) A local staff is used for coding in specific surveys, as when policemen code crimes and school personel code education.

It is clear that decentralized coding is in conflict with other important goals such as decreased respondent burden and low error rates due to non-response.

The error problem in decentralized coding differs in no substantial way from that in centralized coding. However, the possibilities for detecting and correcting the errors are more limited in the former case.

16.1 Respondent coding

16.1.1 Foreign experiences

In Jabine and Tepping (1973) the possibility of using decentralized coding of occupation and industry is discussed. Sometimes one might consider using a schedule that requires the respondent to self-code his occupation and industry major group. Jabine and Tepping also suggest mixed centralized-decentralized strat-

egies in situations where data for both large and small areas are collected. Thus a self-coding check-box procedure can be used in large samples drawn in order to provide small-area major-group data, while a centralized procedure can be used for detailed data for large areas. In this latter case a rather small sample could be sufficient.

First it should be pointed out that with self-coding special care is necessary with respect to questionnaire tactics and the order of items. This subject is dealt with in many books and articles on social research; see for example, Phillips (1973).

In Winch et al. (1969) it is shown that when untrained college students coded their fathers' occupations little quality was lost but the savings in costs were considerable.

In Eckhardt and Wenger (1975) the households of a number of low-performing students at grade levels 1-3 and 7-9 were surveyed. The resulting sample was much more heterogeneous with respect to race, education and income than in the Winch et al. study. The respondents were asked to self-code occupation, among other things. Still, it was found that there was a high correlation between expert coding and the self-coding procedure.

Also, in Taubman (1973) some evidence is given which favors decentralized coding on major group occupation coding, at least when the respondent population consists of highly educated people.

In Bauman and Chase (1974) a study of married females is described in which interviewers coded occupation. White interviewers interviewed white respondents only, while black interviewers interviewed black respondents only. It was found that the white interviewers performed as well as central staff coders

but that black interviewers did not. The study offers no explanation for this difference. The main conclusion from the study is that for the entire sample, central coders were more successful than interviewers when coding occupation.

In Miller (1975) some data concerning the self-coding in the Australian censuses are given. In the 1971 census about 75 percent of the questions were to be self-coded. In a pilot study for the 1976 census three different types of schedules were tested. The schedules contained different proportions of self-coding questions, but each type appeared equally acceptable to the public. It was also found that the greater the amount of self-coding involved, the better the schedule performed with respect to item non-response.

16.1.2 Swedish experiences

In the study of working hours of Swedish teachers (Sveriges Offentliga Utredningar (1971)) decentralized coding was used. The teachers in the sample were given a coding dictionary for a number of different activities, and were to code their activities during a specific period of time. Of course in all statistical studies involving diary keeping there exist certain measurement problems: for instance, problems with telescoping and other memory effects. It was found that memory effects were positively correlated with the number of code numbers available. Further, it was found that the amount of working hours was overestimated by about an hour per week when the teachers themselves coded the activities.

The teachers claimed that decentralized coding was not particularly difficult. However, it was found that they tended to make it easier for themselves at the end of the survey; i.e. they tended to code fewer activities.

We know of one other decentralized system, used by the Swedish army. In that system all males liable to military service are required to self-code their occupation and education in order to make it easier for the authorities to put them into proper positions. The system is updated by mail surveys in which each respondent receives extensive lists of code numbers for the two variables. The codes are represented by alphabetic registers and the respondents are supposed to find their proper categories and write down the appropriate code numbers on the schedule.

This is exactly the kind of system we should like to know something about. Unfortunately the data are used without any controls whatsoever; however the response rate is 90 %, which is a good result for a mail survey without follow-up.

16.2 Interviewer coding

Another possibility is to ask the interviewers to code in connection with the interview. This method has been used in the Swedish Labor Force Survey for many years for coding occupation. We have the error rate studies described in chapter 5, but we do not know how well either the respondents themselves or a central coding staff would perform on the same survey. We are now considering having the interviewer coding of occupation in the Labor Force Survey taken over by the central coding staff, simply because this staff is already coding the industry variable.

16.3 Decentralized coding staff

In Swedish censuses the field work is administered in part by local staffs hired by the authorities in specific geographical districts. There are a couple of hundred such districts, and by tradition these local staffs prepare the census schedules prior to central processing at Statistics Sweden. In the 1975

census it was decided to let these local staffs code some data for economical activity in order to produce important estimates more quickly than possible with the regular census routines. These data were later coded again by the central staff. Thus, by means of an evaluation study we were able to compare the results of the decentralized coding with those of the centralized. Here we have a third type of decentralized coding: decentralized coding staffs scattered all over the country. This decentralized coding concerned the variables relationship to head of household, employment, type of working place and two geographic variables. The comparison, available for the first two variables only, is found in Table 16.1 below.

Table 16.1 Decentralized coding vs. centralized coding in the 1975 Swedish census.

Variable	Error rate (%)	
	Decentralized	Centralized
Relationship to head of household	2	.6
Employment	4.1	.6

The table shows that the error rates differ substantially. Obviously, it was not possible to train and supervise the local staff to the extent necessary. A discussion of the decentralized coding in this census is found in Andersson (1977).

In an agency like Statistics Sweden these types of local staffs are often used in different surveys. In some instances they might be considered professional respondents: for instance, school personnel code some educational data. Another example is the coding of crimes, which is done by police officers at local police stations or by special personnel at some of the larger stations. An error rate study for the decentralized

coding of crimes is presented in Jansson (1978). It was found that the overall error rate was 10.1 percent and for some major groups the error rate exceeded 35 percent. These error rates are considered far too high and have brought about changes in the coding rules. However, the alternative of centralized coding has not been discussed.

16.4 A final remark

Asking the respondents to code themselves can result in considerable reduction in survey costs. Some studies also indicate that major group coding by respondents can have quality as good as centralized coding. Decentralized coding carried out by respondents can in some instances produce more valid answers than those collected by interviewers: see the studies reported in Hochstim (1967), Krohn et al. (1975) and Hinrichs and Gatewood (1967). However, using local staffs of different types, whether police officers, municipal authorities or school teachers, will probably lead to quality problems. Such coding can be viewed as centralized coding with decentralized coding problems. In some of the preceding chapters we have observed how difficult it is to maintain a fairly tight control program. In decentralized coding by staffs this problem will probably increase heavily, since different rules and customs emerge in different staffs. Still, the cost factor is becoming more critical and the whole area of decentralized coding needs to be investigated further.

APPENDIX IV CODING PROCESSES

The discussion in Part II has been on the individual techniques for manual coding. In a practical situation the statistician has to choose between different means, and often some options must be abandoned because of financial or administrative difficulties. The practioner usually has to resort to a limited number of techniques, and this results in coding processes which are far from ideal.

In this appendix, we will take a "global" view: we will discuss the simultaneous use of various techniques for manual coding in Swedish surveys.

IV.1 The case of Sweden - a review

It is stipulated by the Swedish government that almost all Swedish official statistics are to be produced by Statistics Sweden. In 1979 the number of surveys or products exceeded 200. In at least 100 of these some kind of coding is carried out. However, most of the coding is of a rather modest nature; large-scale coding is found in a few surveys only, such as the censuses, the Labor Force Survey, and some other continuous surveys of individuals and households. The extent of coding control at Statistics Sweden during the last decades has not been especially large. The main attempts at control have taken place within our population censuses where, for instance, dependent verification has been used for some time. Population censuses in Sweden have been conducted every fifth year since 1930 (except for 1955 when the census was cancelled for special reasons). Censuses of housing, though, have not been taken regularly until recent years. In 1965 the first coordinate Census of Population and Housing was conducted; subsequent censuses in 1970 and 1975 followed the same pattern. The basic purpose of a census of population and housing is to provide data in the area of living conditions and housing which are not available from other surveys or current statistics. Examples of variables under study are relationship to head of household, age, occupation, location of place of work, type and size of dwelling, equipment, and rent. The system for data collection has changed over the years. Enumerators have not been used in "modern" time. Instead, up to 1970 most of the forms were collected by janitors and house-owners for further delivery to a local scrutiny board (one board in each commune) before the final delivery to Statistics Sweden. During the 1970 census the media took an interest in the data collection methods and their confidentiality was questioned. As a result the 1975 census was carried out as a pure mail survey with no involvment of janitors and house-owners.

During the 1965 census the dependent verification system used in that census was evaluated; this resulted in a partial abandonment of the system for 1970. In the 1970 census independent verification was used at Statistics Sweden for the first time, at least in regular data processing. Unfortunately, the error rate to be controlled by the statistical quality control process was underestimated, and this eventually resulted in financial problems: we could not afford all the rectifying inspection needed. Thus, for parts of the material we had to be content with simple estimates of coding quality. During this census coding was carried out by means of optical character recognition; this facilitated the matching of independent code numbers, since the matching could be done by computer. The coding was thoroughly evaluated and the results of the evaluation studies were used to improve the coding process in the 1975 census. During the 1975 census the coding was conducted by means of special facilities: every coder had a computer terminal on which to key in the code numbers. The terminal facilities permitted a much smoother coding operation than ever before. Independent verification was used throughout the entire operation, coding instructions were stored in the computer (enabling the coder to display, for instance, an alphabetic dictionary of occupations with their associated code numbers on the screen), and coding on schedules or special coding forms were no longer necessary. Because of a tight quality control program and certain changes in the code for industry and occupation compared to previous censuses, the error rates were kept much lower than before.

As to other surveys conducted by Statistics Sweden we must admit that the most common quality control approach is no control at all. It is true that coding errors are taken care of, at least to some extent, by editing and correction before tabulating the data, but such efforts have not involved feedback to the coders. During recent years, though, a growing interest in evaluation studies can be noticed. Examples of such studies

are coding of planned occupation in surveys of pupils, industry in surveys of living conditions, and causes of death. Almost all of the evaluation studies indicate that coding is far from perfection. Many of the problems could be traced to imperfect codes. For instance, different occupation codes are used within Statistics Sweden, i.e. a specific verbal description is coded by x (true code number) in one survey and with y (true code number) in another. This lack of coordination creates a lot of problems and it seems necessary to start eradicating in this jungle of codes. The goal must be one and only one code for each variable under study (allowing of course, coding at different levels). At the same time, however, coding control must compete with control of other statistical operations for financial resources.

IV.2 Coding in population censuses in Sweden

1965 Census of Population

The coding in the 1965 Census of Population concerned five variables. Relationship to head of household, employment, and status were one-digit variables, industry was a three-digit variable; furthermore, for a sample of households occupation, another three-digit variable, was also coded. All code numbers were assigned directly on the schedules close to the underlying verbal descriptions. During the entire coding process coding was controlled by means of acceptance sampling with rectifying inspection of substandard lots. The process started with total control; as soon as a coder met certain quality standards he was transferred to sampling control. The actual AOQL was 5 % with respect to defective schedules rather than defective code numbers: a schedule was classified as defective if dependent verification revealed that at least one of the code numbers assigned by the production coder on that schedule was incorrect. The coding continued for a year and was carried out by some 200 coders. Each coder received a three-day education and train-

ing course prior to the start of production. All coding took place in one building; thus we had genuine centralized coding. The coders were divided in 20 teams with 10 coders in each team. One team member was the crew leader and the other worked as production coders or verifiers. Generally, those coders considered best were used as verifiers. Since the quality control program was modest, it could be administered by the crew leader. One person was acting chief for the whole coding operation. When problems occurred the chief could gather the crew leaders and inform them of different measures taken such as revisions of the coding instructions.

As described in chapter 11 a study of the performance of the dependent verification was conducted where it appeared that the dependent verification had been ineffective.

1970 Census of Population

The 1970 Census of Population was much more extensive than its 1965 predecessor, and the number of variables to be coded increased. The situation was very different from that in 1965 and demanded special procedures in the coding operation. Thus, variables describing economic activity were coded by one staff and variables describing education were coded by another. In all 440 coders were used although they did not all work simultaneously. The coding was split between two cities at a distance of 150 miles. Within each city the coding was further split in different buildings. All together, coding was carried out in seven different places. This was no longer genuine centralized coding: it had a decentralized character, even though the central agency at least formally controlled the operation. In each city about 15 teams were in operation with approximately 14 members in each team. The quality control program was much more extensive than in 1965, and as a consequence each team had its own administrator. The tasks of this person were distri-

buting schedules to coders and verifiers, keeping each coder record in order, and so forth. In fact, the administrator usually did not have even a coder's education. Within each team, coding of education was carried out by a specialized staff.

Code numbers were assigned on special forms which permitted optical character recognition. The main variables to be coded were relationship to head of household, type of activity, occupation, status, industry, kind of employment, type of conveyance to place of work and number of hours at work.

Each coder received two weeks' education and training, much of it by means of master sets. The quality control program was more ambitious than ever before and independent verification was introduced for the first time in a regular production at Statistics Sweden. The control focused on the variables for economic activity. Although the hope was to abandon the dependent verification used in 1965, lack of financial resources forced us to the following control measures. One-third of the schedules were controlled by dependent verification on a total basis, one-third were controlled by two-way sequential independent verification on a sampling basis, and for the remaining third we had to be content with a simple estimate of the coding quality.

A coder's work was independently verified as soon as he produced a lot with fewer than 10 % erroneously coded schedules. During the independent verification a lot was classified as defective if the number of incorrectly coded schedules exceeded the acceptance numbers in Table IV.1 below.

Table IV.1 Acceptance numbers in 1970 Census of Population

Number of schedules in the lot	Acceptance number
< 200	3
200-284	4
> 284	5

Defective lots were rectified by means of dependent verification. This plan had a formal AOQL of 10 % for defective schedules, but since the rectifying inspection was dependent the actual AOQL must have been worse.

The independent verification was not a real success, since the process average error rates were underestimated: too many errors were found and as a consequence too much rectifying inspection was needed. Thus, as mentioned, for approximately one-third of the material we had to resort to independent verification without rectifying inspection due to shortage of time and financial resources.

The coding was evaluated by means of expert coding of a sample of 7 000 individuals.

1975 Census of Population

Our general experiences and the results of the evaluation of the 1970 census resulted in a much smoother coding process in 1975. A series of basic changes created a much better coding environment. Thus, the number of variables was decreased compared to the previous census. The education variables were deleted and all coding could be carried out by one staff. Also, a new technical device was introduced: instead of coding on optical character recognition forms, each coder had access to a computer

terminal (or, rather, a display console). The code numbers were keyed in which facilitated the entire operation and also certain control measures. Among the advantages the following could be mentioned.

i) Verification became a smooth part of the system since the computer handled the matching of different code numbers;

ii) No special verification schedules were necessary;

iii) The computer was able to carry out instant logical checks;

iv) The coder was able to display, for instance, the occupation alphabetic register on the screen whenever he wanted guidance for this specific variable;

v) Coding statistics and record keeping were easily provided.

Each coder had two weeks' education and training before the start of the regular coding. Most of this time was devoted to master set coding. All training was carried out on consoles in a way almost identical to the regular coding.

The unwieldy semi-centralized coding of 1970 was replaced by a more tight setting in 1975. In 1975 the coding took place in one city, one building and, for most of the time, on one floor in that building. As before, coding was carried out by teams. Each team consisted of 12 coders including 2 "experts". In all, 19 teams were used since only 152 consoles were available, coding was carried out in shifts during a period of 15 months.

The verification was a variation of the two-way sequential system used in 1970. The practical details of the system are rather complex with different flows of different items. However, in principle, the system was two-way with expert adjudication of each non-matching case. The best coders were used this time as production coders.

The acceptance sampling approach was abandoned. Instead, a specific volume of verification (25 %) was established, and with this constraint the goal was to obtain maximum quality. To start with, all production coders were totally verified. As soon as a coder produced a lot with less than x erroneously coded individuals his work was verified on a sampling basis with a verification of every zth coded individual. The values of x and z was changed continuously in order to keep the specified verification volume according to plans. Coders producing substandard lots were retrained on the spot.

Before coding began several preventive measures were taken. Among other things, the census schedule was tested from the viewpoint of coding, and the many changes of the 1970 codes resulted in less difficult coding cases.

The coding was evaluated by means of expert coding of a sample of 7 000 individuals (see Andersson (1977)).

More detailed descriptions of the coding in the 1965, 1970 and 1975 Censuses of Population are found in Official Statistics of Sweden (1969, 1974 and 1979).

Part III AUTOMATED CODING

Contents

	17	The challenge
A		THE MAIN COMPONENTS
	18	A bird's-eye view of automated coding
	19	The dictionary
	20	The automated coding procedure
B		EXPERIMENTS AND APPLICATIONS
	21	Non-Swedish illustrations - an overview
	22	Swedish illustrations - industry
	23	Swedish illustrations - occupation
	24	Swedish illustrations - goods
	25	Swedish illustrations - book loans
	26	Swedish illustrations - medicine

17 The challenge

As illustrated in part II, manual coding has various drawbacks. Especially it is: time-consuming, costly, error-prone and boring.

To cope with these drawbacks, it appears inevitable to focus on the very basis of manual coding and to consider the possibilities offered by access to a computer of developing a basically new approach. This idea is, of course, not in principle new: for instance, at the US Bureau of the Census geographic coding has been conducted by means of computer since 1963. What is new is the suggestion in that agency that the computer be used extensively in the coding of such complex variables as occupation and industry. This suggestion may be viewed as a natural extension of earlier uses of computers in the editing operations.

During the last decade we have conducted a series of experiments in Statistics Sweden in order to find out whether or not it is possible to automate the coding process. It is quite clear that these experiments should in some cases have been analyzed in much more detail. However, since our aim has been to investigate if coding can be handled by the computer at all we chose to conduct many experiments at the price of less detailed analysis[1].

In part III, we will review these Swedish endeavors.

In section A, we discuss the basic methodology of automated coding; more specifically we present the main components of a system for automated coding. These components are the construc-

---

[1] It should be pointed out that the errors occurring in manual coding have many different causes both within the coding process and in earlier stages in the production process. Thus, making the coding more effective can imply changes in other stages of the production process, such as data collection and editing. Such changes are not treated in this thesis.

tion of a dictionary, the entering of descriptions into the computer, the matching between descriptions and the dictionary, the coding, and the evaluation.

In section B, we consider applications. Thus chapter 21 sums up the main non-Swedish experiments in this field. The summary covers experimentation with variables such as underlying causes of death, occupation, and industry. The remaining five chapters deal exclusively with Swedish illustrations covering five different variables, namely industry, occupation, goods, book loans, and medicine. Each illustration is presented with respect to experiments, applications and evaluation.

A THE MAIN COMPONENTS

18 A bird's-eye view of automated coding

Before we enter upon a technical discussion of the main components of automated coding, we present an admittedly very simple example. We distinguish four operations:

i) Construction of a computer-stored dictionary;
ii) Entering element descriptions into the computer;
iii) Matching and coding;
iv) Evaluation,

and will discuss them in turn.

18.1 Construction of a computer-stored dictionary

In automated coding a dictionary stored in the computer takes the place of the coding instructions used in manual coding. Obviously the construction of such a dictionary is a very important task. The construction work could be carried out manually but, when dealing with multi-digit variables, using the computer seems to be a better alternative. The resulting dictionary should consist of a number of verbal descriptions with associated code numbers. The descriptions could be a sample from the population to be coded or a sample from an earlier survey of the same kind. Of course an important problem is the size of the sample underlying the dictionary construction. Whether the dictionary is constructed manually or by computer the code numbers appearing in it should be those assigned by the best of the available coders with previous special independent verification procedures.

We will consider here the following dictionary for the variable occupation:

Occupation description	Code number
Bicycle repairman	586
Guard	901
Hairdresser	415
Night watchman	908
Sailor	736
Statistician	311
Store detective	902
Supervisor	618

18.2 Entering element descriptions into the computer

A survey of N = 7 elements has yielded the following verbal descriptions (observations) with respect to occupation:

Element no	Description
1	Hairdresser
2	Statistician
3	Sailor
4	Worker at a service station
5	Chief of the ambulance corps, Ann Arbor
6	Supervisor
7	Saylor

These descriptions are to be entered into the computer. One possible method is to punch the descriptions in a more or less free format on cards or magnetic tape. However, this method has some serious drawbacks: first it consumes a lot of "space", and second, the errors involved in large-scale keypunching of alphabetic information are relatively unknown; moreover such keypunching is rather costly.

A better alternative would be to have the verbal information directly available for optical character recognition. Unfortunately the recognition of hand-written letters is not yet sufficiently developed for this purpose.

There are reasons to believe that at present the entering of verbal descriptions to the computer is the most important practical problem in designing systems for automated coding.

18.3 Matching and coding

Each element description now put into the computer is compared with the list of occupation descriptions in the dictionary. If an element description agrees with an occupation description (is a "match"), it is assigned the corresponding code number; otherwise it is referred to manual coding. The outcome is summarized below:

Element no.	Assigned code number	Referred
1	415	
2	311	
3	736	
4		x
5		x
6	618	
7		x

We will elaborate on this operation.

In the example above, only four element descriptions had exact counterparts in the dictionary. It is necessary to find methods which make it possible to code different variants correctly, including spelling errors. In our example we have one such case: sailor is obviously misspelled as "saylor" (element no. 7).

General matching problems and algorithms for matching have been dealt with by several authors; see Tepping (1968), Newcombe et al. (1959) and Sunter and Fellegi (1967). The matching theory presented in those papers has relevance for automated coding but, as we have mentioned above, special problems occur in automated coding. It is obvious that the routines for searching must be as efficient as possible. One way to do this is to design subdictionaries to which reference is triggered by certain information in the first part of a word or the first word in a phrase. Another way is to permit partial coding for variables whose coding system is built on the Chinese box principle. In O'Reagan (n.d. a) some general aspects of this kind of matching are given.

According to Palme (1971), a natural way to organize a system for computerized coding is to make it heuristic. The system could be designed in such a way that its continuous evaluation and revision could be carried out by means of decisions of a pool of coding experts. Then the design of the dictionary keeps pace with the collection of expert information. To reduce the amount of time for searching, another type of subdictionary could be designed on the basis of information on the most recent input or calculations of different input frequencies. For instance, one could design "quick" dictionaries consisting of the most recent, say, one thousand input words or phrases with associated code numbers or the one thousand most common input words or phrases with associated code numbers. If the input information coincides with some description in this quick dictionary, then a code number is assigned to the element. If not, an extended search in the ordinary dictionary is called for.

It is also possible to simplify the input by means of some formatting procedure, e.g. the removal of vowels or the removal of those parts of input words which carry no discriminating information.

In an automated coding system we will obtain exact matching for a fraction of all elements only. A primary task in developing such a system is to design criteria for the degree of similarity between input words and dictionary words necessary for them to be considered to match.

18.4 Evaluation

The system must include continuous evaluation studies. Such studies aim at

i) controlling the quality of computerized coding;
ii) improving the dictionary and;
iii) controlling the cost.

In our example only four of the elements were coded by the computer - the rest had to be referred to manual coding. Whether this is economical or not is a question to be answered by the evaluation. Are the referred cases more difficult to code than those taken care of by the computer? In our example the description of element no. 4, at least, is vague and there is probably a need for some special care on part of the manual coder. Does the dictionary need improvement? These are other questions to be resolved by evaluation.

19 The dictionary

There are two general kinds of algorithms for automated coding: weighting algorithms and dictionary algorithms. Weighting algorithms assign weights to each word-code combination using information from a basic file: when a new record is to be coded the program chooses the code number which is assigned the highest weight for the specific record word. Dictionary algorithms look in a dictionary for words or word strings which imply specific code numbers: when a new record is to be coded the program determines whether the word or word string matches any word in the dictionary. If no match occurs the record is rejected and referred to manual coding.

At the US Bureau of the Census a number of different algorithms have been developed and investigated during the last decade. In some straightforward applications like the geographic coding automated coding has been quite successful. Recent efforts deal mainly with the more complex coding of occupation and industry. Four algorithms are described in Lakatos (1977 a, b). Two of them, the O'Reagan and the Corbett algorithms, are dictionary methods. The remaining two, the IMP and the INT algorithms, use the weighting method. The INT algorithm is due to Rodger Knaus, is further described in Knaus (n.d., 1978 a, b, 1979).

In Statistics Sweden we have worked with the dictionary approach only. Thus we have nothing to add with respect to other algorithms.

A record in the dictionary looks roughly like this.

Identification number	Verbal description	Code number

Thus the computer-stored dictionary is a parallel to the dictionary and the coding instructions used in manual coding. In order to create such a dictionary a number of operations must be carried out:

i) Choice of basic material;
ii) Sampling a basic file from the basic material;
iii) Expert coding of the basic file;
iv) Establishing inclusion criteria for dictionary records;
v) Construction of a preliminary dictionary;
vi) Testing and completing the preliminary dictionary.

In this chapter we give examples of possible ways of carrying out these operations.

19.1 Choice of basic material

If we are going to test or introduce automated coding in a survey we must have access to the basic working material; the most suitable basic material is the set of schedules in that specific survey. The best result is obtained if this basic material is the one coded by the computer but naturally this is rarely possible - time is not on our side. Instead the basic material must often consist of

i) material from an earlier survey of the same kind; or
ii) material from a pilot study; or
iii) material from another kind of survey in which the same variable was included.

It is important that the basic material be up to date. In the Swedish experiments with automated coding on census material the basic material consisted of schedules from the 1965 censuses. On the basis of that material different 1970 and 1965 census materials concerning industry and occupation were coded by

computer. We found that the coding of the 1965 material was more successful than the coding of the 1970 material. A contributing reason for that is that a change in the population occurred over these five years. Changes can be structural; e.g. entry and exit of industry and occupation denominations, which occur frequently. Also it is possible that the reporting pattern has changed over such a long period of time. One example could be the following. In the 1965 Census of Population, cleaners described their occupation as "cleaner". In the 1970 census a new term, "local keeper", was used by some cleaners. That term had not existed in 1965 and as a consequence was not represented in the basic material. The result was that the dictionary based on the 1965 census material was not able to code 1970 census individuals describing their occupation as "local keeper".

As pointed out earlier the conditions for automated coding are very propitious in censuses: the amount of data is enormous, hundreds of coders must be hired and trained, manual coding is time-consuming and the quality problems are substantial. In a census it should be possible to use as basic material a combination of i) and ii) above. Basic material as in iii) should only be used in exceptional cases, since the reporting pattern for a certain variable could differ substantially between different surveys due to, for instance, different data collection methods. For instance, in one of our studies the occupation variable in the Labor Force Survey was coded by computer by means of a dictionary based on census material. The degree of coding was rather poor partly due to the fact that in the census the respondents themselves described their occupations while in the Labor Force Survey an interviewer wrote down a description obtained from the respondent. As a consequence the descriptions of occupations obtained in the Labor Force Survey are in general much longer than those obtained in the census.

19.2 Sampling a basic file from the basic material

From the basic material we must sample a number of records in order to construct a dictionary. These records constitute a basic file with which we work at the final construction stage. The sampling of records could be carried out in different ways, for instance

- a simple random sample of records
- a controlled random sample
- a subjective sample

With the first approach, descriptions with low frequencies have a small probability of being included in the file. This is generally not a negative consequence.

The second approach could be realized by means of stratification of the basic material. If it is known that the pattern of descriptions varies a great deal within certain parts of the basic material and varies very little within other parts, it is desirable to use a disproportionate sample constructed by sampling proportionally more records from the parts which vary more.

The third approach can be used when we must consider special cost situations or when we have access to different kinds of subjective prior information.

The size of the sample is a problem, irrespective of the kind of approach we use, since each description should be coded by experts.

In the experiments with automated coding carried out at the US Bureau of the Census, a very large initial random sample of records was chosen: sample sizes of about 100 000 records have been used. In the experiments in Statistics Sweden the basic file has consisted of at most 14 000 records. Despite

that, evaluation studies show comparable results. Possible explanations are that a few code numbers are generally sufficient to code a large portion of the records and that the Swedish language is less complex (at least in this context) than English.

Suppose that we have a basic material of N records. A verbal description is associated to each of these N records. Thus it is possible to define a population of unique verbal descriptions

$$\{X_I\} = X_1, X_2, \ldots, X_M$$

where $M \leq N$

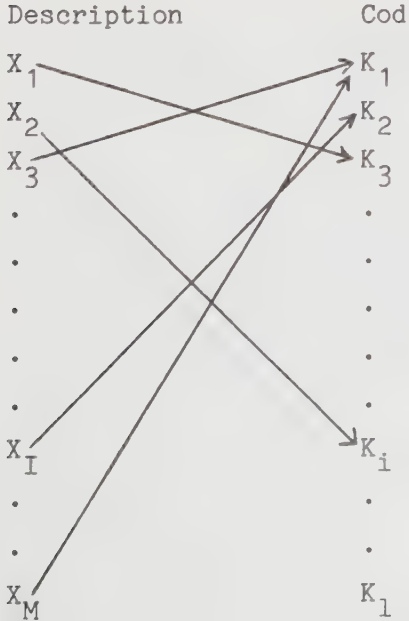
The descriptions occur with relative frequencies

$$f(X_1), f(X_2), \ldots, f(X_I), \ldots, f(X_M)$$

respectively, where

$$\sum_{I=1}^{M} f(X_I) = 1$$

Furthermore, suppose that to every verbal description a code number can be assigned according to some scheme, for instance:



Let us state the actual problem.

When constructing a dictionary, how should a basic file (a sample of records with verbal descriptions) be chosen in order to obtain a certain degree of coding, P, when coding dictionary-independent material, under the constraint that the proportion of correctly coded elements Q is higher than some prespecified value Q_v?

We can distinguish at least the following cases:

A. Every (X_I) and $f(X_I)$ is known in the basic material to be coded. This problem is trivial. We merely pick J descriptions (where $J<M$) from the set $\{X_I\}$, resulting in

$$\sum_{I=1}^{J} f(X_I) = P$$

provided that $Q > Q_v$. These J descriptions constitute our dictionary.

If the frequencies $f(X_I)$ are arranged in a decreasing order we obtain a minimum-volume dictionary.

B. Every (X_I) and $f(X_I)$ is known in basic material similar to but not identical with the material to be coded.

This situation may occur in a repetetive survey when we have access to information about $f(X_I)$ and (X_I) for a survey carried out at time t and want to code the elements in a forthcoming survey at time t + 1. Obviously there is no need for scientific sampling. However, it might be a good idea to pick L descriptions such that

$$\sum_{I=1}^{L} f(X_I) > P \quad \text{and } Q > Q_v$$

in order to compensate for changes in the response pattern between t and t + 1.

C. The different X_I and $f(X_I)$ are more or less unknown in the basic material.

This is the most common situation. Information about the descriptions must be obtained before the dictionary can be constructed. A possible method is to choose an initial sample (sample 1) of n_1 elements with associated verbal descriptions. The size of the sample is determined by budget restrictions, since nothing is known about the different $f(X_I)$. By means of this sample (basic file) version number one of the dictionary is constructed. (The methods for construction are discussed later.)

Now a new sample (sample 2) is drawn consisting of n_2 elements with associated verbal descriptions. These elements are coded automatically by means of version number one. If the resulting degree of coding is higher than P and the proportion of correctly coded elements is higher than the prespecified value Q_v, then we can, for instance

i) use version number one as our dictionary, or
ii) take a new sample (sample 3) consisting of n_3 elements.

This sample is coded by means of version number one. If P and Q_v once again are smaller than the resulting coding degree and the resulting proportion of correctly coded elements respectively, then we can use version number one as our dictionary.

If the values P and/or Q are not reached in the test coding of sample 2 and/or sample 3 then, for instance, new dictionaries based on sample 1 + sample 2 or on sample 1 + sample 3 can be constructed.

This is trial and error. However, it is a possible way of proceeding if we wish to avoid one very large initial sample as a basic file.

The main question, however, remains and unfortunately we cannot offer any good solutions: what sizes should we choose for the different samples? Of course, budget restrictions is a simple criterion and results in, for instance, $n_1 = n_2 = n_3$ where $n_1 + n_2 + n_3$ is decided by budget restrictions. It is also possible to establish sample sizes by means of a prior distribution which approximates $f(X_I)$. Some facts which should be decided in advance are

i) the number of categories
ii) the frequencies of different code numbers in earlier surveys
iii) the code numbers for certain words or word strings.

Another possibility is the following. Two independent samples {A} and {B} of the same size ($n_A = n_B = n$) are drawn from the basic material of N elements. The verbal descriptions generated in this way can be represented by a sampling distribution for each of the two samples. Now we can experiment with some kind of overlapping measure. For example, we can compute the frequency with which descriptions occur in both samples. Suppose that this number is n_x; i.e., n_x elements among the 2n elements have descriptions that occur in both samples. Then those elements remain which have descriptions occurring in only one of the samples. Denote the frequencies for these elements n_a and n_b respectively. Thus

$$n_x + n_a + n_b = 2n$$

A very simple measure of the overlapping is

$$n_x/2n$$

Such a measure has no desirable characteristics other than being a first aid in the sample size determination. When it exceeds a certain value (determined from prior information from earlier studies), then we terminate at size 2n. Otherwise, we have to start all over again with larger initial sample sizes.

It is also possible to sample in the following way. A frequency limit f_o is chosen in a subjective way and a sample of n elements is drawn and coded. If a certain description j in the sample occurs with frequency x_j where $x_j \geq f_o$, then that certain description is included in the dictionary, provided that the description is associated with a specific code number in at least T % of the cases; otherwise it is not. Then a sampling distribution is constructed in which the descriptions are ordered according to decreasing frequency. Such a distribution often resembles the following figure.

In Norberg (1976 a, b, 1977) the sample size problems in our experiments at Statistics Sweden are discussed. Among other things we studied four random samples of coded occupation descriptions. In the population to be coded it is possible to define a parameter P_j, the proportion of elements with description j in the population. If we take a simple random sample of n elements with replacement from this population, the resulting distribution for "number of elements with description j", x_j, is binomial with parameters n, P_j.

Now let

$$y_j = \begin{cases} x_j \text{ if } x_j \geq f_o \\ 0 \text{ otherwise} \end{cases}$$

where $x_j \geq f_o$ is the dictionary inclusion criterion.

$\Sigma\, y_j/n = r(n)$ is the expected proportion of elements in the population to be coded by the dictionary. $\Sigma\, y_j$ was calculated for four different sample sizes (n = 200, 400, 800, 1600) and a number of different f_o. Since we were interested mainly in predicting what would happen with r(n) for large sample sizes by observing $\Sigma\, y_j$ for small sample sizes, we were interested in how r(n) increases when n increases for different values of f_o, and in how r(n) behaves when the increase of f_o keeps pace with an increasing n?

In Diagrams 1-3 below the four sample sizes and the eight f_o-values (2, 3, 4, 5, 8, 16, 32, 64) give rise to 32 points which are connected. In Diagram 2 the sample sizes are registered on a log scale.

Diagram 1

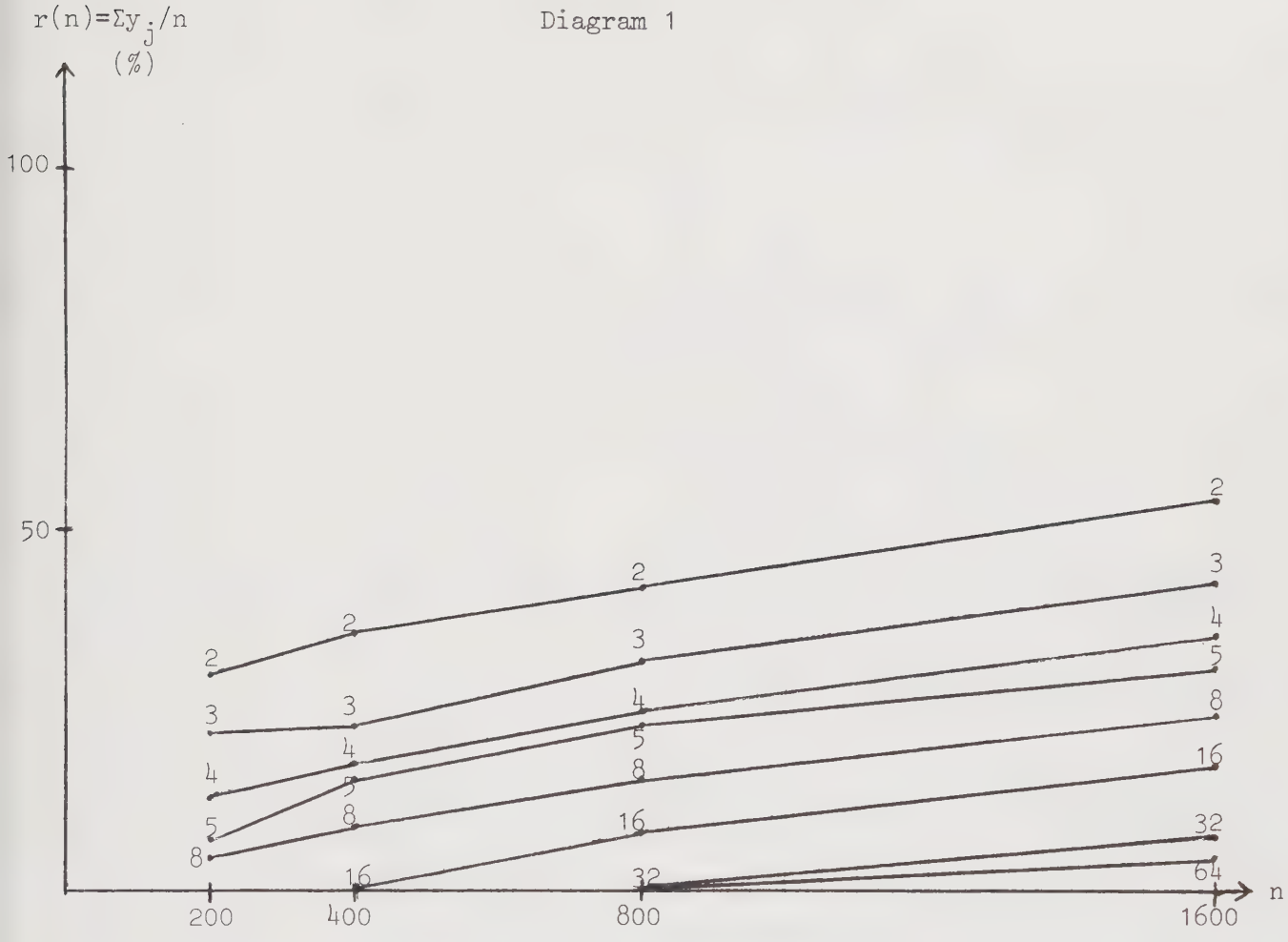


Diagram 2

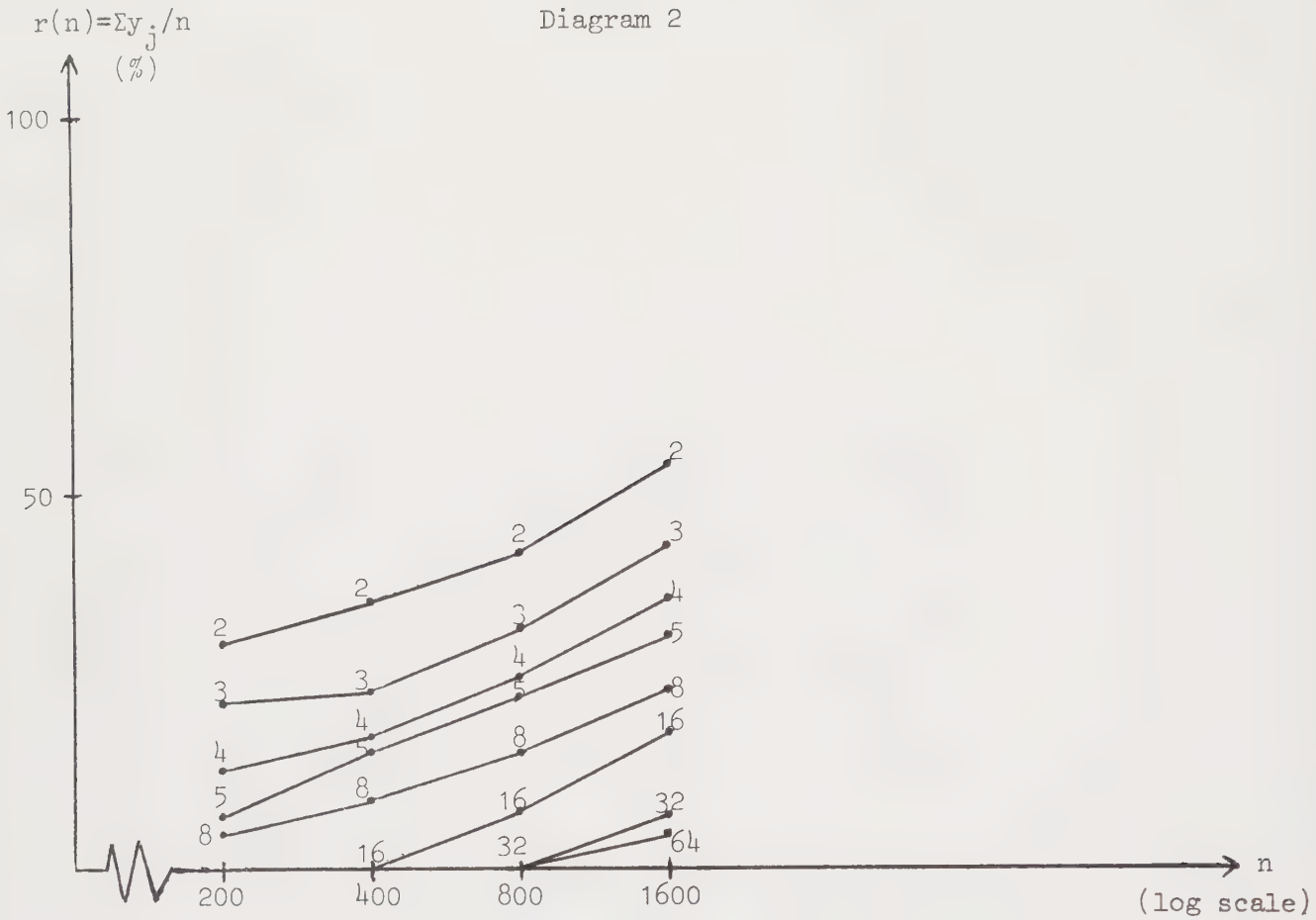
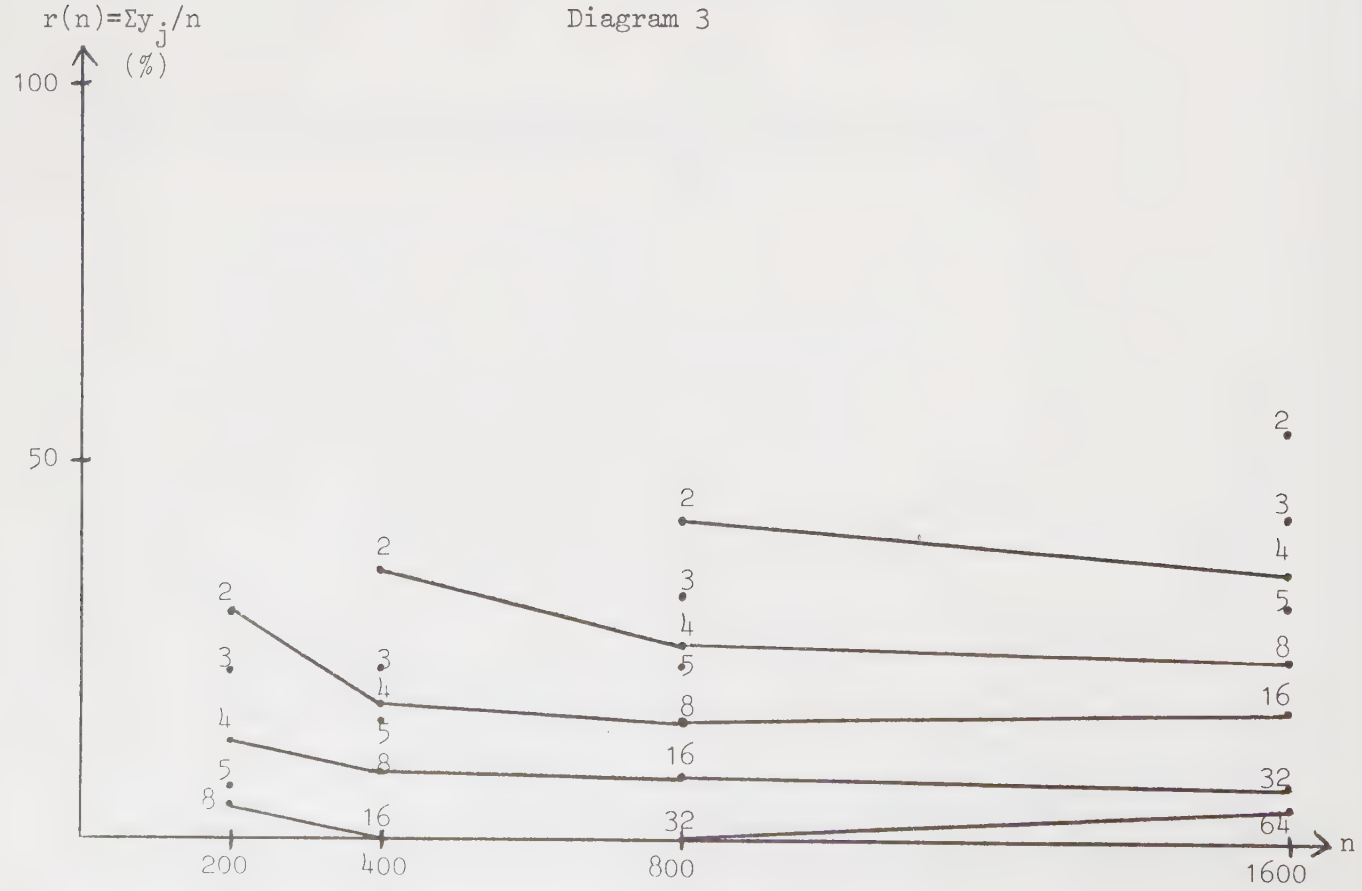


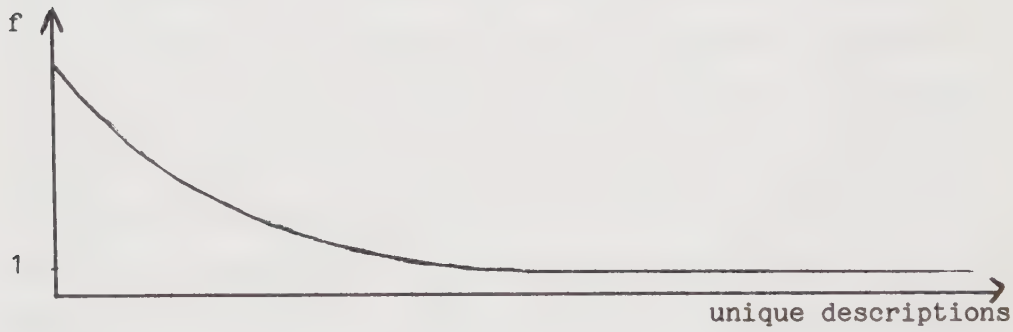
Diagram 3



The curves in Diagram 1 are remarkably parallel which means that the proportion of descriptions with frequencies 2, 3, 4 etc. does not depend on the sample size. The curves in Diagram 3 are almost parallel to the n-axis for n between 800 and 1600, at least for f_o-values exceeding 4.

The important feature of these curves is that with them one can predict, for instance, r (6400) for f_o=8 by observing r(1600) for f_o=2.

Let us finally get down to business. A typical frequency diagram for unique descriptions is the following:



The typical diagram has a very straggling tail provided that the descriptions are ordered with respect to the frequencies with which they occur. In fact, most unique descriptions occur only once or twice. Fortunately, it is usually the case that a very small part of the code numbers available is sufficient to code a large part of the records, and this fact is very important. In O'Reagan (1972) a closer look revealed that, for one variable, 7 % of the code numbers could handle 50 % of the records.

Thus by means of a rather small initial sample it is possible to get a decent dictionary. Our experiences show that vast increases of the basic file do not add much with respect to coding degree. We have to concentrate on, say, the 15 % most frequently used categories and accept manual coding of most of the remaining part.

19.3 Expert coding of the basic file

In order to construct a good dictionary the basic file has to be coded with high quality, and for this work we have to use the best coders available. We can call these persons expert coders. Experiments show that not even these people code correctly with probability one: on the contrary, we have observed that census coders, some with more than ten years of coding experience, can produce error rates of 10 % for variables such as occupation and industry. Thus the expert coding must be carried out in conjunction with a control operation.

As a consequence the expert coding becomes a financial factor in a system for automated coding. Schemes for independent verification have already been discussed in chapter 12. Here we merely point out that in a control situation of this particular kind, some schemes which are less attractive from a production point of view can nevertheless be possible solutions. The two-way sequential independent scheme is the best solution in ordinary statistical control of the coding operation. There are reasons to believe, however, that a four-way independent scheme without a majority rule, for instance, may produce better results for coding a basic file.

19.4 Establishing inclusion criteria for dictionary records

After the expert coding is completed we have access to a basic file in which each record consists of a verbal description together with a code number. The verbal descriptions can be classified into different categories:

a) Descriptions of high frequency which all point at some specific code number;
b) Variants of descriptions of type a);
c) Descriptions of low frequency which all point at some specific code number;

d) Descriptions of high or low frequency with which different code numbers are associated;
e) Variants of descriptions of type d).

When constructing a dictionary we are aiming at a prespecified coding degree of P of which Q_v should be correctly coded. This goal cannot be specified until production has started. But to be able to come close to the P-and Q-values we must, at the time of dictionary construction, specify certain criteria for including a description in the dictionary. This problem has been discussed above but from a different viewpoint.

In principle, all descriptions pointing at some specific code number should be included in the dictionary. Whether this can be done in practice depends on how large a dictionary we can accept, and this in turn is a function of the searching time of the matching program. If the searching time is independent of the size of the dictionary, then all descriptions pointing at specific code numbers should be included. Otherwise we must define what is meant by "high frequency". This decision depends on sample size and number of categories among other things; for instance a small enough sample generates no highly frequent descriptions at all. A simple piece of advice is to have a low value of the concept "high frequency", for instance ≥ 3, since it is always easier to remove than to add descriptions to the dictionary.

The descriptions to be included should all point at specific code numbers. In those cases in which a certain description is associated with different code numbers we should check with the expert coders whether this is possible or if an erroneous coding has occurred when coding the basic file. In the latter case we have no problem: we merely include the description together with its correct code number. However, the former case is not unusual in a practical coding situation. A certain

description can generate different code numbers for different elements depending on the code, the coding instructions, and what supplementary information the coders use. This problem can be dealt with in two different ways. One alternative is to omit such descriptions from the dictionary. The second is deliberately to permit a certain degree of erroneous coding when using the dictionary. The probability for this misclassification should be smaller than $(1 - Q_v)$ to allow a small margin of erroneous coding of variants.

The choice of Q_v should be guided by our experience with regular manual coding: Q_v should be chosen to be in the neighborhood of what an expert coder is expected to manage. In normal manual production coding the quality obtained is somewhere between the rates for the production coder and the expert coder, since most of the coding is handled by production coders and the expert coder is called in only for matters of verification, completion and so forth. There are reasons to believe that those descriptions which the automated coding system fails to code are difficult ones, generally speaking. Thus Q_v should be kept in the neighborhood of the rates for expert coders as a pure act of compensation.

A dictionary can contain records with descriptions in categories a), b) and c) above. Including records in categories d) and e) results in bad coding quality. We should aim primarily at including descriptions in categories a) and c). After that we consider how many variants should be included. The response pattern of the elements shows no lack of variation, at least for such variables as occupation and industry. If we want a P-value of some magnitude, a number of variants must be included in the dictionary one way or the other. Of course the most simple way would be to include all variants in the basic file pointing at some specific code number. But the variants are of low frequency (otherwise they would not be variants); i.e. their frequencies do not reach the high-frequency criterion.

That means that the pattern of variants depends on the actual sample, and another sample would produce other variants. In this situation a possible solution is that the computer be programmed to recognize discriminating word strings. In the ideal situation one such word string represents many variants of a certain description. We have used this idea in some of the experiments with automated coding at Statistics Sweden. The simplest way is to construct a separate subdictionary for the variants which is scanned after the primary dictionary is checked.

One example of the content on these two types of dictionaries is the following. The basic file contains the description

BILMEKANIKER 752 (CAR REPAIRMAN)

Possible variants of this description are BILMEK and BILMEKANIK, two common abbreviations. It seems suitable to include the word string BILMEK in the subdictionary of variants. First, however, we have to check that BILMEK does not appear as a word string in some other word included in the primary dictionary. For instance, the word string ÄGMÄST is contained in both JÄGMÄSTARE (FORESTRY OFFICER) and VÄGMÄSTARE (ROAD-BUILDER, CHIEF) and should not be included in the subdictionary if the latter words are parts of the primary dictionary.

Generally speaking, the risk of misclassification increases with the number of variants included.

19.5 Construction of a preliminary dictionary

A dictionary can be constructed by man or by computer. Presumably a combination of the two is the most effective. In our studies at Statistics Sweden this work has mainly been carried out manually. However, we now have a program for computerized dictionary construction.

19.5.1 Manual construction

The expert coded file is first sorted as to code number (list no. 1) and after that alphabetically (list no. 2). These two lists form the basis for the dictionary construction. List no. 1 is used to get some hints about the structure of the verbal descriptions sorted under a certain code number. Now, consider the following example of the construction of a dictionary of occupation code numbers. In the basic file the following descriptions can be found under code number 908.

908

Description	Frequency (f)
Vaktmästare (Guard)	5
Nattvakt (Night watchman)	4
Portvakt (Porter)	4
Vaktassistent (Assistant guard)	3
Väktare (Watchman)	3
Garagevakt (Garage watchman)	2
Parkordningsvakt (Park watchman)	2
Bevakningsman (Patrol man)	2
Idrottsplatsvaktmästare (Sports-ground watchman)	2
Föreståndare (Superintendent)	1
Förman (Foreman)	1
Tillsynsman (Supervisor)	1
Industriportvakt (Porter in industry)	1
Vaktman (Watchman)	1
Saktman (Hatchman (punching error))	1
Vaktmästare tillsyn och skötsel av sportkläder (Guard, supervision and attention of sports clothes)	1
Trafikvakt (Traffic officer)	1
Ordningsman (Commissionare)	1
Kontrollant (Controller)	1
Butikskontrollant (Store detective)	1
Bevakningstjänsteman (Patrol officer)	1
Stationschef vid ambulansen, Borås (Chief of the ambulance corps, Ann Arbor)	1
Skötsel av idrottsplatsen och sporthallen (Attendant at sports-ground and gym)	1
Skolbespisningsvakt (Watchman for school dining-hall)	1

Thus, in the basic file there are 42 elements with code number 908. We now choose a frequency limit f_o for coding of "high-frequency" descriptions. Let us choose $f_o \geq 3$. This means that the descriptions

Vaktmästare
Nattvakt
Portvakt
Vaktassistent
Väktare

are stored in the preliminary version of the primary dictionary which is scanned first in automated coding. The primary dictionary is called PLEX.

After that we start looking for discriminating word strings. An example of such a set is

Vakt
Bevak

These word strings are stored in a secondary dictionary. This secondary dictionary, called SLEX, is scanned if PLEX fails to code.

If the specified preliminary dictionary descriptions are good enough then 33 of the 42 "908-elements" in the file can be coded. If this is a satisfactory result remains to be seen. Some experiments can throw some light on that matter.

Now what is meant by "good enough"? When we turn to list no. 2 (not reproduced here), we realize immediately that "vaktmästare" is not a good description since most "vaktmästare" have been assigned code numbers others than 908. As a consequence "vakt" will not be a proper discriminating word string. "Nattvakt" has in five cases out of six been assigned 908 and could be allowed to remain, if we are willing to accept a certain coding error (misclassification). "Portvakt" has in four cases out

of seven been assigned 908 and in the other three 931; it obviously cannot remain. "Vaktassistent" is a unique description and can remain, as can "väktare". Now the PLEX-dictionary for code number 908 has been reduced to

Nattvakt
Vaktassistent
Väktare

and the SLEX-dictionary to

Bevak

Since a number of highly frequent descriptions were lost because of their lack of unequivocalness, we are now in a position to reconsider the inclusion of low-frequently but unequivocal descriptions in the PLEX-dictionary. A check shows that all descriptions with frequency 2, viz. "garagevakt", "parkordningsvakt", "bevakningsman" and "idrottsplatsvaktmästare" are unequivocally coded to 908. These can be included in PLEX if we wish a higher coding degree.

The word strings in SLEX, which are more expensive to look for, should be common to several descriptions or be parts of special highly frequent descriptions. Therefore it is possible to enlarge the SLEX for code number 908 with, say, "nattva", "vaktas" and "väkt".

Now we have to be sure that those word strings included in SLEX do not fit the PLEX descriptions for other code numbers. In addition, they must be unique in the sense that a given word string does not appear more than once in SLEX. Unfortunately, these matters cannot be checked until a first version of the dictionary is available for each code number.

Thus our preliminary dictionary for code number 908 looks like this:

PLEX

Nattvakt
Vaktassistent
Väktare
Garagevakt
Parkordningsvakt
Bevakningsman
Idrottsplatsvaktmästare

SLEX

Bevak
Nattva
Vaktas
Väkt

The main problem with SLEX is that we do not know in advance how it behaves when new records are coded. We are in fact facing a loss of information when we include words in SLEX, since SLEX identifies a code number less specifically than PLEX does. This loss can be measured for alternative sets of SLEX words. In Dolby (1977) and O'Reagan (1968 b) the concept of information loss in similar contexts is treated in detail.

19.5.2 Computerized construction

The manual work described above can be carried out by a computer, at least to an important extent. Here some different approaches are presented.

19.5.2.1 O'Reagan's algorithm

In O'Reagan (1972) the following approach is discussed. Let us assume that we have only three categories in the code. We have a basic file consisting of seven elements with associated verbal descriptions. The file looks like this:

Element	Description	Code number
1	AB	1
2	ABCK	1
3	KWQ	1
4	BQ	2
5	WR	2
6	QM	3
7	MM	3

The letters represent word strings. We see that word string A only exists in descriptions coded into category 1. K and C, too, seem to indicate code number 1. R indicates code number 2 and M code number 3. A first approximation is

Element	Word string	Indicated code number
1, 2	A	1
2, 3	K	1
2	C	1
5	R	2
6, 7	M	3

If we eliminate these informative word strings and look at what remains we obtain the following:

Element	Remainder	Code number
1	B	1
2	B	1
3	WQ	1
4	BQ	2
5	W	2
6	Q	3

B together with Q indicates code number 2; B without Q indicates code number 1, etc. Now we have:

Element	Word strings	Indicated code number
1, 2	$B\overline{Q}$	1
3	WQ	1
4	BQ	2
5	$W\overline{Q}$	2
6	$Q\overline{WB}$	3

This list is combined with the first-approximation list. As a result we obtain a dictionary for coding new elements. Deviations from the algorithm are permitted by using the same type of criteria that we discussed above: the "parameters" N, T_1 and T_2 are introduced.N is the number of times a word or a word string occurs in the basic file before it is included in the dictionary. T_1 is the criterion for unequivocalness; if T_1 is less than 1 we buy coding degree at the price of misclassification risk. ("Nattvakt" above was an example of this.) T_2 is a criterion for incompleteness. Let us assume that there are two kinds of seamstresses, those working in tailoring businesses and those working on ready-made clothes. If, in the sample, 17 % of the seamstresses work in tailoring businesses and 83 % make ready-made clothes, then an element for which we have only the information "seamstress" is coded as a ready-made seamstress if T_2 is less than .83.

19.5.2.2 Corbett's algorithm

This algorithm is described in Corbett (1972), Owens (1975 a, b) and Bostrom (1975). It starts with the creation of the basic file (which is already assumed a priori in the O'Reagan algorithm). This process has three different steps.

- First a sample of elements is selected. Each element is represented by a record; each record contains information about several variables together with associated code numbers.

- Each industry description is marked with an asterisk to make possible for the computer to distinguish between occupation and industry descriptions.

- For each variable the words of each description are sorted alphabetically, and then duplicate words are removed.

Now the search for so-called classifiers can start. A classifier is a word string (at least one word) which satisfies the following conditions:

- it shows up often enough
- when it shows up, a specific code number has been assigned to it in at least R % of the cases.

The dictionary is built in the following way.

a) All one-word strings with code numbers are listed, and among these a certain number of classifiers are identified. These classifiers constitute the first chapter of the dictionary.

b) The identified classifiers are removed from the basic file.

c) All two-word strings with code numbers are listed. The classifiers among these constitute the second chapter of the dictionary.

d) These classifiers also are now removed from the basic file.

e) All three-word strings with code numbers are listed, etc.

.
.
.

k) The dictionary is ordered.

When the automated coding is carried out the dictionary is searched chapter by chapter.

19.5.2.3 Computerized construction in Statistics Sweden

We have already mentioned that our present dictionary construction system at Statistics Sweden generates a dictionary with two chapters, PLEX and SLEX. Our experience shows that it is

rather easy to construct a PLEX manually, but that manual SLEX construction is much harder to manage. Thus we have made a program for computerized construction of SLEX. (As a consequence a computerized PLEX is obtained as a simple special case.)

We have tried a few different versions of the program. Some of the earlier versions are described in Lyberg (1976) and in Gustavsson and Karlsson (1977). The present version, a package called AUTOCOD, is described in Gustavsson and Karlsson (1978), Karlsson (1978) and in Bäcklund (1978). All programs are written in PL1. AUTOCOD contains routines for

- the creation of computer stored dictionaries (PCLEXK)
- the coding of descriptions (PCAUTOK)
- the updating of dictionaries (PCLEXUP)
- the evaluation of dictionaries (PCLEXT)

PCLEXK creates a PLEX and a SLEX. The procedure involves three steps. The program LEXLADD creates space for a possible SLEX. LEXKONS creates PLEX and SLEX. For each PLEX description, say, a six-character abbreviation starting with the first character is tested for inclusion in SLEX. If that abbreviation fits another PLEX description it is rejected and a new abbreviation is created starting with the second character of the PLEX description. The procedure is repeated at most six times; if no valid abbreviation is obtained the procedure goes on to the next PLEX description. Finally LEXLIST lists the dictionaries by means of EASYLIST. Parameters that can be varied include

- possible use of a list of prefixes which, when making a dictionary of, say, goods, removes such word strings as pounds, roll, and pairs
- minimum frequency f_o (the inclusion criterion)
- tolerated degree of equivocalness
- minimum length of words in SLEX

PCAUTOK codes new records by means of PLEX and SLEX. PCLEXUP is used when we want descriptions to be removed from or added to an existing dictionary. PCLEXT is used to evaluate a dictionary when we have access to a material with manual code numbers assigned.

PLEX and SLEX can be updated simultaneously or separately.

20 The automated coding procedure

Suppose we have a computer-stored dictionary at our disposal. Verbal descriptions of elements are entered into the computer. The descriptions are matched against the dictionary and if match occurs a code number is assigned. One of the problems with automated coding is that, because of the almost infinite number of possible input variants, exact agreement between element descriptions and dictionary descriptions is obtained only in a fraction of all cases. Our present matching program deals with exact matching only. One could argue that a matching and coding program should also be able to handle some spelling and keypunching errors. While this can be done, experience shows that such program ingredients are time-consuming and therefore costly.

When we go from exact character-by-character matching to more sophisticated inexact matching procedures, we find that common properties between the automated coding problem and other problem areas have already been studied in the literature. The problem can be viewed, for instance, as having features in common with noisy channel coding theory, i.e. error-detecting and error-correcting codes. If a received code word differs only a little from a valid word it is corrected to the nearest valid word. The automated coding process could also be recognized and developed through a self-adaptive or learning process. This can be accomplished by showing the computer a sufficient number of expertly coded elements. Some references within this area are Kay and Martins (1970) and Kellerman (1972). The problem can also be identified as a record-linkage problem, and the literature within this area is extensive. General record-linkage problems are dealt with, for instance, in O'Reagan (1968 a), Tepping (1969), Bachi et al. (1967), Sunter (n.d.), IBM (1969), Jaro (1972), Coulter (1977) and Du Bois (1969). Some parallels can also be found in the automatic recognition of continuous speech and the understanding of languages. These topics are

discussed in Dixon and Tappert (1971), Tappert (1972), Tappert et al. (1973) and McCalla and Sampson (1972). Similar problems have also been studied in the fields of data clustering and taxonomy.

In the two weighting algorithms for automated coding mentioned in chapter 19, pattern recognition ideas were in fact used. The dictionary algorithms deal mainly with simpler forms of matching. However, in Statistics Sweden we have considered such concepts as similarity between words and different measures of distance.

20.1 Similarity between words

Explicit measures of similarity between words are rather scarce in the literature. In Estabrook and Rogers (1966) a very simple one is presented, but unfortunately it is not especially relevant to automated coding.

Implicit measures exist, if by "implicit" we mean applying a rule for correcting a misspelled word to one which can be found in the dictionary. The point is that the similarity is so great that a correction is justified.

Methods of searching for and correct spelling errors can be classified into at least three groups: methods for abbreviation, pairwise comparison of letters, and obtaining prior probabilities concerning build-up, frequencies and erroneous spelling of words.

20.1.1 Methods based on abbreviation techniques

One possible means for decreasing the risks of spelling errors is to automatically abbreviate words, removing those letters which are most error-prone. Such letters is identified from a frequency table. It is also possible to remove letters in certain positions which are judged to be highly uninformative.

Such an abbreviation method is described in Blair (1960). Prior to abbreviation all letters are given points according to error risk and position. Those letters which have the highest score (risk score + position score) are removed. It is assumed that the dictionary is abbreviated as well. After that matching is carried out. Blair's point system looks like this.

Risk points

Letter	Points	Letter	Points
A	5	N	3
B	1	O	4
C	5	P	3
D	0	Q	0
E	7	R	4
F	1	S	5
G	2	T	3
H	5	U	4
I	6	V	1
J	0	W	1
K	1	X	0
L	5	Y	2
M	1	Z	1

Position points

Position	Points	Position	Points
1	0	9	5
2	1	10	5
3	2	11	6
4	3	12	6
5	4	13	6
6	4	14	6
7	5	15	6
8	5	16-w	7

The tables are based on empirical studies. Obviously such studies would give different results for different languages; thus Blair's tables - based on English - could not be used in applications in Statistics Sweden.

In Davidson (1962) a method for matching misspelled names against a dictionary containing correctly spelled ones is given. The method has been applied in flight reservation systems. With this procedure, names are abbreviated to five characters where the first character is always the first letter of a person's surname, the fifth character is always the first letter of his first name (if that information is available), and the remaining positions contain suitable surname consonants. In Jackson (1967) another abbreviation method for coding of company names can be found.

The reason why the abbreviation methods work so well is semantic redundancy inherent in language: the words we use are "unnecessarily" long.

20.1.2 Methods based on pairwise comparison of letters

Methods based on positionwise comparisons suggest themselves. Here the concept of "distance" between words is used; usually this distance is some kind of Euclidian distance. Consider the following. Let us assume two elements e_i and e_j, which could be words (input and dictionary) with the corresponding column vectors

$$\underline{x}_i = \{x_{i1}, x_{i2}, \ldots, x_{ip}\} \quad \text{and}$$

$$\underline{x}_j = \{x_{j1}, x_{j2}, \ldots, x_{jp}\}$$

The Euclidian distance between the vectors is

$$d_{ij} = \sqrt{(\underline{x}_i - \underline{x}_j)' (\underline{x}_i - \underline{x}_j)} = \sum_{v=1}^{p} \sqrt{(x_{iv} - x_{jv})^2}$$

If we let the vector components be letters and let an agreement between positions be indicated by 0 and non-agreements with 1, then the expression above is reduced to number of 1s. Now look instead at the following examples

Word to match	STATISTICIAN
Dictionary word	STATISTICIAN
Distance	000000000000

The number of 1s is 0 and as a consequence the distance is 0 too.

Word to match	HIARDRESSER
Dictionary word	HAIRDRESSER
Distance	01100000000

The number of 1s is 2 and thus the distance is 2. Now a criterion of the following kind could be applied. "An input word at a distance of not more than x from a dictionary word is assigned the code number of that dictionary word."

Examples of problems arising with this kind of similarity measure are the following.

Example 1

Word to match	TAILOR
Dictionary word	SAILOR
Distance	100000

The number of 1s is one; if the critical value is greater than one and the word "TAILOR" is not a member of the dictionary, then we are stuck with a coding error. The solution to this

sort of problem is to apply different weights to different positions. One possibility is never to permit deviations in the first position if this position is considered especially informative from a semantic point of view. However, such decisions must be based on linguistic considerations.

Example 2

Word to match	CUTLER	CUTLER
Dictionary word	BUTLER	SUTLER
Distance	100000	100000

A given input-word can be at the same distance from several dictionary words. Obviously, in this case the distance rule cannot be applied without serious problems.

The simple measure of distance used above is only one of many alternatives. In Statistics Sweden we have experimented with a rather special measure, built on Spearman's rank correlation coefficient. In this technique, input words are compared to dictionary words by means of the rank correlation coefficient.

Consider the following example.

Example 3

Word to match	SUPERINTENDENT
Dictionary word	INTENDENT

Step 1: Letters of the "word to match" which are not found in the dictionary word are removed. We start from the left. The letters S, U and P are removed since they cannot be found in the dictionary word. E is found and constitutes the first letter in a "revised word to match". R cannot be found and is thus removed. The letters I, N, T, E, N, D are found and constitute the next six letters of the revised word to match.

The next letter, E, in the word to match is found but all the E's are already "occupied". Thus this E is removed. Finally the letters N and T are found; they constitute the end of the revised word.

Thus we obtain two words with equal length:

Revised word to match EINTENDNT
Dictionary word INTENDENT

Step 2: Rank the nine letters with the dictionary word as a basis. The ranking is

Revised word to match 2 3 4 1 6 7 5 8 9
Dictionary word 1 2 3 4 5 6 7 8 9

Step 3: Compute the rank differences. These differences, d_i, are

+1, +1, +1, -3, +1, +1, -2, 0, 0

respectively.

Step 4: Compute Spearman's rank correlation coefficient R:

$$R = 1 - \frac{6 \sum_{i=1}^{n} d_i^2}{n^3 - n}$$

$$= 1 - \frac{6 \cdot 18}{729-9}$$

$$\approx .85$$

By means of a criterion, $R > R_o$, it is decided whether a match has occurred or not.

As pointed out in Davidson (1972), however, the big question is how the program would know in the first place to try to match "superintendent" to "intendent". It will try all other words such as "supervisor", "supernumery" and even, say, "attendant" or "printer". One would need study a fairly large number of examples before reaching a judgement on the efficiency of the technique or similar ones.

In Damerau (1964) another method of pairwise comparison is discussed. In this system an input word is compared only with the set of dictionary words which differ in length from the input word by at most one letter. For instance, an input word of seven letters is compared to dictionary words consisting of six, seven or eight letters. A match is said to have occurred if

- the words are of equal length and only one letter does not match;

- the words are of equal length and two consecutive letters do not match, but a transposition of these letters in the input word creates identical words;

- the input word has one more letter than the dictionary word and a removal of the first unmatching letter makes the words identical;

- the dictionary word has one more letter than the input word and a removal of the first unmatching letter makes the words identical.

In Morgan (1970) a spelling correction algorithm is used as a two-stage structure:

First stage: From the dictionary words a subset is selected which contains all words of which the input word may be a misspelling.

Second stage: To each word of the subset, a pairwise comparison algorithm is applied which determines whether or not the input word is a misspelling of that dictionary word.

In Muth Jr and Tharp (1977) an automatic method for correcting spelling and typing errors from teletypewriter keyboard input is proposed. The correcting process uses a heuristic tree search.

In Brodda and Karlgren (1968, 1969) the distance measure is discussed in connection with trademark similarity checks. There it is argued that simple distance measures only give rough estimates of the intuitive similarity. A modified procedure in which different weights are given to different parts of a word or a syllable is called for. This procedure should be based on linguistic evidence in order to separate the important differences from the unimportant ones.

20.1.3 Methods based on obtaining prior probabilities concerning buildup, frequencies and erroneous spelling of words

Starting from prior tables for the existence of letter combinations, erroneous spellings and frequency of words (specific to each language) for each word in a dictionary, the posterior probability for that word being erroneously spelled is computed. Such methods are discussed in Thorelli (1962), Vossler (1964) and Cornew (1968).

20.1.4 Comments on similarity between words

If erroneous spelling and variants do not exist, automated coding becomes a rather trivial problem from a technical point of view. One possible approach is to accept only exact matchings and let the rest be coded manually. When we first considered automated coding at Statistics Sweden we thought that permitting

exact matchings only would produce large lots of uncoded material and thus result in zero or even negative payoff. As will be seen later, this is not necessarily true. We have, for instance, applications in which the verbal information is already key-punched, and even a rather modest degree of automated coding is most valuable. The coding of goods is a similar example: here the coding is carried out with few errors, but the manual coding operation is so costly and time-consuming that a degree of coding in the area of 60-70 % is worth consideration.

However, in other situations we might have to work with some kind of similarity measures. For each application there exists a critical value for the coding degree to be economically acceptable. As mentioned above, exact matching by means of a PLEX can be sufficient in some situations. Sometimes, though, we have to apply some of the procedures described above; in fact SLEX is the result of such efforts. SLEX is used with exact matching but the construction is based on ideas on similarity.

In Statistics Sweden we have experimented with distance and correlation measures, but as things have turned out we have experimented with PLEX and SLEX more and more and ultimately often used PLEX only.

20.2 Evaluation and control

A final and necessary step in an automated system is evaluation and control. This operation must be continuous and must run in parallel with the coding process. Its primary goal is to make necessary alterations in the process; i.e., to make changes in the dictionaries in order to maintain the prespecified level of accuracy.

The coding degree and the error frequency are the main characteristics studied in a control operation. Both are used during the dictionary construction and it seems natural to use them during the control procedures too. The coding degree, p, is defined by

$$p = \frac{n}{N}$$

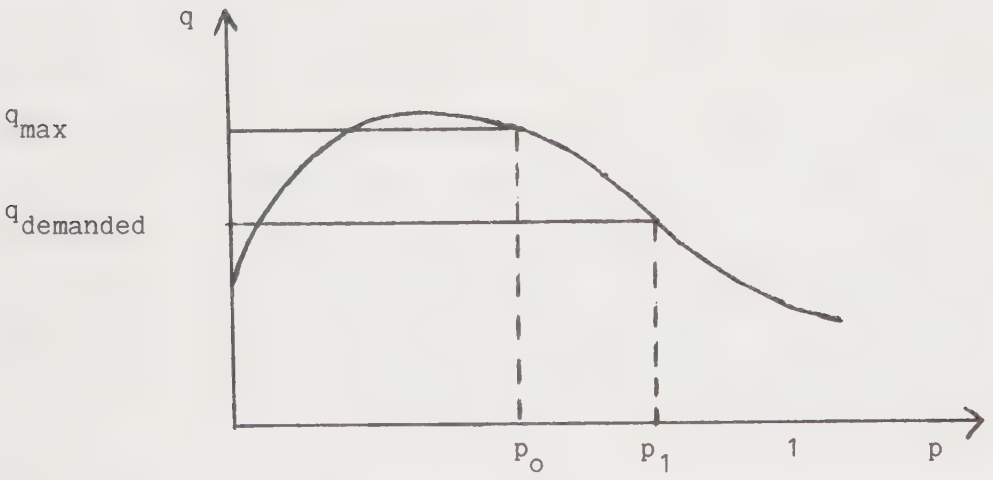
where N is the number of elements entered into the computer and n is the number actually coded by the computer. This measure is relatively unreliable in the sense that a low value does not necessarily imply that automated coding does not pay off, nor does a high value imply that automated coding is profitable. The coding degree must be related to the quality if it is to be a useful measure.

Suppose that m out of the n coded elements are correctly coded. Then the proportion

$$1 - \frac{m}{n}$$

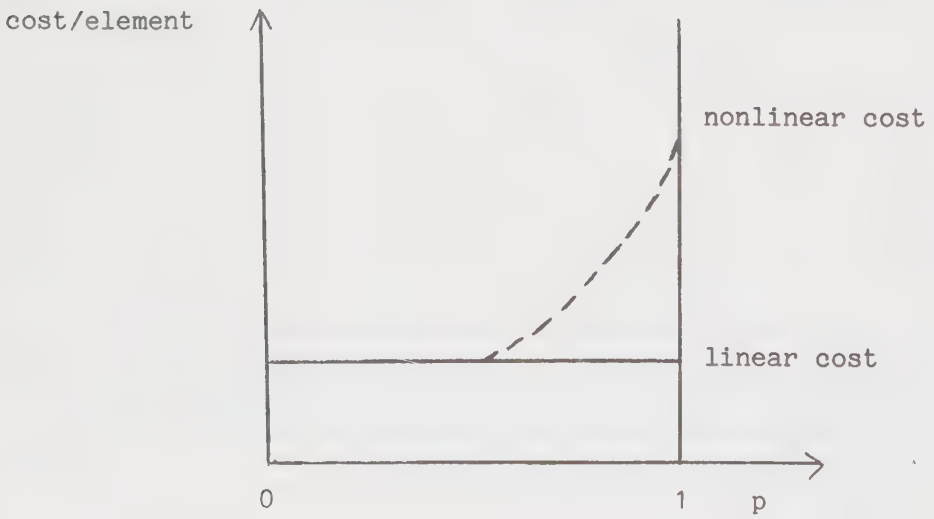
is the error frequency. Obviously it is more important to have a large value of m/n = q than a large value of p. When comparing different experiments, the product pq might be helpful. Unfortunately even this measure must be used with great care, since the combination p = .5, q = .9, say, is much better than p = .9, q = .5. One should strive primarily for a q-value as high as possible, i.e. as good a coding as possible. After that one can concentrate on increasing the coding degree p. This proportion could be increased until q starts to decrease. It is even possible to increase p to the price of a reduction in q, but then the payoff must clearly outweight the loss in quality.

By operating with different p- and q-values it is possible to obtain an approximation of the optimal size of the proportion 1-p, i.e. the proportion of the elements which must be coded manually. Consider the following figure.



If every verbal description costs about the same to code manually irrespective of its complexity, then p_o is some kind of optimal coding degree. If the desired quality is $q_{demanded}$, a value less than q_{max}, then the optimal coding degree is p_1.

Depending on the matching program and the dictionary construction, we might end up with a cost function other than the linear one implicitly assumed above. For instance, use of the rank correlation coefficient is associated with a nonlinear cost function (computer costs), as in the following figure.



If a cost function for automated coding is similar to that outlined above, then of course we must take this into consideration when we assess the values of p and q. Thus the costs for coding might establish a natural limit for the coding degree, i.e. an optimal value of p.

Now, what role does the cost for manual coding of the proportion 1-p play? It plays an important role in calculating the costs of the entire coding operation, including both automated and manual steps. The descriptions which the computer is unable to code can be more complex than those it did code. Besides, there is a higher fixed cost associated with the manual coding of the proportion 1-p and furthermore the manual code numbers must also be keypunched. These costs must be considered when estimating the total coding cost. However, from a strictly economic point of view (neglecting losses due to decreasing quality), we should carry on with the automated coding as long as the cost per element is lower than that for manual coding.

When the q-value is estimated, before the possibilities of automated coding is fully examined, one should decide upon a suitable level of ambition. Should the computer act like a regular coder or like an expert coder? Should a combination of PLEX and SLEX or some other type of dictionary be used? Before deciding this a manual dictionary construction can be tested against a computerized one, say, for some specific code numbers.

The goal of continuous control is to check whether the quality of coding is kept on the pre-specified level. A secondary goal is to gather information that can be used as a basis for changes in the dictionaries and the matching programs. The p- and q-values are estimated continuously. Samples are drawn from the production and the samples are coded by verifiers according

to a suitable independent scheme. These code numbers are compared to those produced by the computer and the resulting q-values can be estimated. The results can be displayed in a table of the following kind.

Computer code numbers (=C) / Manual verification code numbers (=M)	M = C		M ≠ C			
				Which code number is correct?		
	Total	%	Total	M	C	None
Total	n_1		n_2	n_{21}	n_{22}	n_{23}

If q, estimated by $(n_1 + n_{22})/(n_1 + n_2)$, does not meet quality standards the sample must be analyzed. What types of descriptions have been erroneously coded and which have not been coded at all? If n_{23} is relatively large, the sample might be extreme in some sense. Are special sections of the code difficult for the computer?

Of course this analysis could be carried out even if the actual q-value meets the prespecified standards. However, according to the theory of process control, adjustments should generally be carried out only when the process is out of control or in danger of becoming so.

B EXPERIMENTS AND APPLICATIONS

21 Non-Swedish illustrations - an overview

In this chapter a review of some non-Swedish studies of computerized coding is given. It should be pointed out that a few of them have already been mentioned in passing.

21.1 Geographic coding

The systems for geographic coding used in the 1963 and 1967 economic censuses conducted by the US Bureau of the Census are described in Minton (1968, 1969), O'Reagan (1967 b, 1968 a, n.d. b). The system used in the 1972 census is described in Silver (1976). The dictionary is an address reference file consisting of street names and house number ranges within place, census tract, zip code together with code numbers, combined with a city reference file containing post office names with alternate names and spellings, and zip code. After formatting and input check the automated geocoder program is used for matching and coding. One evaluation shows an estimated error rate of 9.3 %.

Also at the US Bureau of the Census the ADMATCH system has been developed (see US Bureau of the Census (1972 a)). The system is intended as a research tool for those who want a fast and accurate system for geocoding a large variety of files.

Geographic coding is also carried out at the US National Center for Health Statistics.

21.2 Underlying causes of death

At the US National Center for Health Statistics (NCHS) underlying causes of death are coded by computer. The classification system is a modified version of the ICD (International Classifi-

cation of Diseases). To process cause-of-death information accurately is a very complex task. This fact inspired NCHS to develop a computerized system for selection of code numbers. Special instructions in the form of decision tables were stored in the computer, and these instructions serve as a dictionary. The system, called ACME (Automated Classification of Medical Entities), is described in Templeton and Evans (1970) and in Israel et al. (1972).

Full-scale application of the system began with the 1968 file of death records for the United States. The evaluation results from that year show that the automated coding degree was 92.6 %. Of these automatedly coded records an estimated 96.7 % coincided with the manual coding. The manual coding error rate was 1.6 % and thus the remaining 1.7 % was produced by the computerized system.

In the United Kingdom (see Weatherall (n.d.)) it has been tested whether data coded by means of a World Health Organization code (which differs from the ICD) could be used as input to NCHS' ACME. However, this coding was found inefficient.

21.3 Techniques for recognizing names

In the United States "Spanish surname" is often used as a surrogate variable. Spanish minorities are sometimes referred to as "Spanish-surnamed". In an attempt to illuminate the correspondence between individuals with Spanish surnames and individuals belonging to Spanish cultural groups, some computerized techniques were compared. Several procedures using different dictionaries of Spanish names were tested. The survey is described in Giesecke (1973).

In Lynch and Arends (1977) some experiments concerning the best surname coding technique are described. According to the authors, the "best" technique can be defined as one which

- places all variations of a given surname in the same category;
- limits the size of the categories;
- creates categories that contain few dissimilar surnames;
- requires minimal processing time and costs.

Five name-coding techniques were examined and compared as to the properties listed above. The technique considered best was used later on in the Statistical Reporting Service Record Linkage System of the US Department of Agriculture.

21.4 Content analysis

Frisbie and Sudman (1968) describe some experimental work at the National Opinion Research Center in the United States on the use of computers in coding natural-language responses in surveys. The dictionary construction and the matching, a modification of the so-called General Inquirer system, is described in more detail in Sudman (1967). It is shown that computerized coding can be equivalent to manual coding when coding free-response information. For instance, the coding of three types of worries resulted in 84 % agreement between manual and computerized coding. The disagreements were evaluated and the errors were exactly the same for the two procedures. The authors state that the costs of an automated system might be a drawback if the system is not used to its full potential, but the automated system offers greater flexibility than the manual system.

In Iker and Harway (1965) a method is given which describes the structure of free responses of the kind discussed above. The application area is content analysis in psychotherapy.

21.5 Occupation and industry

At the US Bureau of the Census the main interest in developing automated coding, at least for the time being, concerns the variables occupation and industry. The different algorithms

developed and tested so far, mentioned in chapter 19, are documented in O'Reagan (n.d. a, 1967 a) and in Lakatos (1977 a, b).

These papers give many experimental results. The most up-to-date results concern an evaluation study carried out in 1976. The test material was a sample of 127 000 coded records, one-fifth of which were used as a test file and the rest were used for dictionary construction. The test material had been verified by a two-way independent system with dependent adjudication for differences.

The O'Reagan algorithm described in chapter 19 was tested with varying parameter values. The best result was obtained when using a minimum frequency limit $f_o = 3$, an "unequivocal" portion $T_1 = 95$ % and with $T_2 = 80$ %. The parameter T_2 has to do with the situation in which an input record matches a proper subset of the words in the dictionary. Suppose that the word "nursery" occurs 100 times in the basic file (that part of the test material which is used for dictionary construction). In 80 of these occurrences it is accompanied by the word "school" and is coded 8211. An input record that says "nursery" with no other information would be coded to 8211 if T_2 is less than 80 %.

For industry coding the best coding degree was 71.5 % with an 87.6 % agreement with manual coding. The speed rate was 2100 records processed per minute. The corresponding rates for occupation were 64.8, 83.5 and 1500 respectively.

The Corbett algorithm was never tested on the full file but it performed worse than the O'Reagan algorithm when direct comparisons were possible. The weighting algorithm IMP produced a coding degree of over 99 % for industry. However, the quality was below that produced by the O'Reagan dictionary algorithm

since the agreement rate reached only 75.6 %. Also, the actual coding was much slower, with 500 records processed per minute. CPU time for construction of the O'Reagan dictionary slightly exceeded one hour.

Changes in the weighting functions improved the agreement rate to 77 % for IMP, but decreased the speed to 450 records per minute. Different manipulations of the coding decision rules in the IMP algorithm increased the quality but the price in decreased coding degree was far too high.

Automated coding of occupation is used in the Central Bureau of Statistics in Denmark (see Danmarks Statistik (1976)). This system performs extremely well because the system itself decides the code; i.e. during the dictionary construction process the decision on how the code should be built is based on the empirical pattern of descriptions. With this technique many of the problems encountered above vanish. The system was established in 1970 when 50 000 unique descriptions were extracted from the Danish Population Record. These were divided into 500 categories and as a consequence it was decided that description i should always be coded in category j. Obviously such a reverse system works smoothly. All dictionary updating is carried out by means of the information contained in income-tax returns.

Automated coding of occupation is also being considered in United Kingdom. In Boston (1977 b, 1978) some preliminary calculations are made on the characteristics of such a system.

There also exist plans for computerization of job-matching operations in the US Federal-State Employment Service system in the 1980s. There already exists a Handbook of Occupational Keywords published in US Department of Labor (1978) which reflects three years of pilot studies in automated job matching.

22 Swedish illustrations - industry

22.1 Experiments

The very first experiment with automated coding (see Lyberg (1972 b, 1974 a, 1977 a, b)) concerned the industry variable. Our dictionary was based on word frequencies; i.e. the dictionary consisted of those words which occurred at least three times in the basic file. The descriptions were chosen manually. When storing the dictionary in the computer, four different programs were used, two written in ASSEMBLER and two in COBOL. The ASSEMBLER programs abbreviated the verbal descriptions of the dictionary and created a dictionary file for serial access. This file then was used as an input to the COBOL programs which created a dictionary file for direct access. The matching part of the system consisted of three ASSEMBLER programs, one main program and two subprograms: one of the subprograms abbreviated the words to be coded while the other one looked for word strings.

The verbal descriptions to be coded were keypunched and subsequently read by the main program. After that the dictionary file was searched for direct access. If matching occurred the verbal description was printed on a list together with the associated code number. Then the next description was read. If matching did not occur a search for a characteristic word string in the verbal description was carried out (by "characteristic" we meant a word string associated with a certain code number in the dictionary). Such characteristic word strings were sorted in the computer by means of serial search of a table. The table was space-consuming but the technique seemed fast and efficient.

If matching occurred during the scanning for word strings then the procedure was equivalent with the direct-access method. If matching did not occur some special matching rules were applied. In this third phase a match was said to have occurred if:

a) at most two letters (positions) were different when an input word was compared with a dictionary word of equal length; i.e. the distance was not allowed to exceed 2. However, the initial letters had to be identical if match was to be considered at all.

b) Spearman's rank correlation coefficient between a dictionary word and an input word exceeded .85; here the word lengths did not need to be equal.

This very first experiment gave the following results. Out of 419 input verbal descriptions 213 were coded or about 50 %. Of these, 170 or 80 % were correctly coded when compared to verified manual census coding. The test also included certain time studies of the different matching routines. The following standardized computer times were observed.

Matching routine	Standardized time
File matching for direct access	2
Matching by scanning of word strings	3
Matching by distance rule	9
Matching by rank correlation coefficient	38

The table shows that most of the computer time was used for the special matching rules. The matching programs were very time-consuming in this first trial, and were eventually shortened substantially.

Now some changes in the abbreviation program were carried out. The descriptions were sorted according to special information in each description. Thus we had an opportunity to divide the dictionary into four subdictionaries. The presence of

a) the word strings factory, industry or manufacture in a description implied the use of subdictionary number 1;

b) the word strings sale, retail, trade or store in a description implied the use of subdictionary number 2;

c) the word strings repair, workshop or service in a description implied the use of subdictionary number 3;

d) none of the word strings listed under a-c above in a description implied the use of subdictionary number 4.

This new system increased the coding degree from 50 to 71 %, partly due to incestuous coding.

Automated coding of industry has also been carried out in an experiment with data from the Labor Force Survey (LFS). Here the subdictionaries mentioned above were not used. Instead we used one PLEX dictionary containing 531 descriptions and one SLEX dictionary containing 400 descriptions. A total of 712 verbal descriptions of industry were entered into the computer. The computer was able to code 65 % of these, but only 69 % were "correctly" coded (in agreement with the LFS manually assigned code numbers). This result was worse than those of the previous census experiments. However, the reason is obvious: in the LFS the interviewers collecting the data strive for detailed descriptions, and as a consequence the descriptions are sometimes composed of one or two whole sentences. In the censuses the respondents themselves fill in the answers and this usually results in short descriptions. The experiment showed that out of the 315 coded descriptions only 50 % were coded by means of PLEX. Usually a large majority is coded by that dictionary, but this time many SLEX words fitted the long input descriptions. For census data we have also tested a computerized dictionary based on 6 000 descriptions in the basic file. This time the descriptions came from the 1970 census, the computer coded 61 %, and 83 % of these were correctly coded.

The main experiments are summarized in the Table 22.1.

Table 22.1 Automated coding of industry.

Experiment	Type of dictionary	Survey	Coding degree (%)	Quality (% agreement)
1	Manual	1965 Census	50	80
2	Manual	Labor Force Survey 1974	65	69
3	Computer-ized	1970 Census	61	83

Perhaps one can accept the low coding degree, but the errors are too frequent. However, we have not been working with this variable very long. In fact, almost all the "trial and error" work, in our opinion the very essence in developing methods for automated coding, is still waiting to be done for the industry variable.

22.2 Applications

No applications have been carried out.

22.3 Evaluation

No specific evaluation studies have been carried out.

23 Swedish illustrations - occupation

23.1 Experiments

When we first started to deal with the occupation variable we had the results from the first industry experiments at hand. Thus we were convinced that the size of the dictionary must be quite large to start with. The basic file consisted of 14 000 verbal descriptions from census material which were coded using an independent verification scheme. The final dictionary, constructed manually, consisted of 900 descriptions. This first dictionary was a type of PLEX dictionary, but the different matching methods (scanning, rank correlation coefficient) could be used if necessary. The first experiment, which was carried out with an independent set of 3 800 occupation descriptions from the 1965 census, gave some encouraging results: the coding degree was 62 % and 95 % of these were "correctly" coded. This was considered to be very satisfactory.

Later on several trials were carried out and we introduced both PLEX and SLEX. With PLEX and SLEX together we sometimes obtained coding degrees around 80 % and qualities around 90 %.

We also tested our program for computerized dictionary construction on census descriptions. An experiment with this resulted in a coding degree of 69 %; 87 % of these were correctly coded. This dictionary is still "untouched by human hands" but is obviously good raw material for further work.

Furthermore, we have constructed a dictionary for LFS occupations. When coding occupations in LFS we do not use the same dictionary as in census coding (as we do when coding industry). We were merely curious to investigate how well the census dictionary would work in LFS coding, and thus, the census 3-digit code numbers were simply translated into LFS 2-digit code numbers. This was done manually and resulted in a PLEX with 1131 descriptions and a SLEX with 393 descriptions.

Thus, 711 independent descriptions were entered into the computer. Of these 81 % were coded and 81 % were in accordance with the LFS coding. This result is worse than those obtained in the census coding experiments. The reason is obvious. A direct translation is not to be recommended due to the different data collection methods: occupation in censuses and occupation in the LFS are not really the same variables. A special LFS dictionary should be constructed by means of a basic material consisting of LFS descriptions.

During 1978 a new set of experiments on LFS material was conducted. This time the computerized construction program was used on a basic file consisting of 6000 descriptions. From these descriptions the program created a PLEX containing 1637 descriptions and a SLEX containing 1230 discriminating words. Three sets of independent new records were coded by means of PLEX or a combination of PLEX and SLEX. Between each run certain changes were made in the PLEX dictionary. In a fourth run a greatly extended dictionary was used.

First run

1002 verbal descriptions were coded. The coding degree was 84.3 %. Of these 84.6 % was in agreement with the manual code number assigned in the LFS. The dictionary was PLEX and SLEX. When only PLEX was used the coding degree decreased to 69 % while the agreement rate increased to 92.2 %. Obviously SLEX was not very accurate.

Second run

Prior to the second run some minor changes were made in PLEX: 22 descriptions were removed and 162 were added. Then a new set of 1000 records was coded by the modified PLEX; the resulting coding degree was 69.1 % with an agreement rate of 93.3 %.

The cost of coding 1000 occupations was 18 kronor (approximately 4 US dollars).

Third run

Prior to the third run some further minor changes were made in PLEX: 19 descriptions were removed and 145 were added. Again only PLEX was used for coding of 989 new records. The resulting coding degree was 69.2 % with an agreement rate of 91.8 %. An evaluation of the disagreements revealed that the absolute automated coding error rate was 4.7 %, which we think is a good result.

Fourth run

The minor modifications of PLEX had no effect on the coding degree. However, the manual coders had an alphabetic list of approximately 11000 occupations at their disposal. This list, provided by Statistics Sweden, is not based on knowledge of empirical response patterns. We were anxious to know what would happen if that list, already available on tape, was used as a PLEX. Said and done. We made new runs of the materials used in runs 1 and 2 and obtained the coding degrees 40.2 % and 36.0 % respectively. When we merged the alphabetic list with our computerized constructed PLEX the coding degree increased to 75.8 % and 74.2 % respectively and the agreement rates increased to 93.0 % and 93.8 % respectively.

The cost of coding 1000 descriptions by means of this large dictionary was 127 kronor (approximately 30 US dollars).

The main experiments are summarized in the Table 23.1.

Table 23.1 Automated coding of occupation.

Experiment	Type of dictionary	Survey	Coding degree (%)	Quality or agreement rate (%)
1	Manual	1965 Census	62	95
2	Manual	1970 Census	66	92
3	Manual	1970 Census	74	84
4	Manual	1970 Census	80	90
5	Manual	Labor Force 1974	81	81
6	Computerized (PLEX + SLEX)	1970 Census	69	87
7	Computerized (PLEX + SLEX)	Labor Force 1976	84	85
8	Computerized (PLEX)	Labor Force 1976	69	93
9	Computerized (PLEX)	Labor Force 1976	69	92
10	Computerized and manual combined	Labor Force 1976	74-76	93-94

We think that it is possible to code occupation in our surveys by means of the program package together with some manual efforts in the dictionary construction phase. However, for an automated system to be considered, the cost situation must be carefully evaluated.

Our experiments with occupation coding are described in, for instance, Lyberg (1972 a, 1974 a, 1977 a, b, 1979).

23.2 Applications

No applications have been carried out.

23.3 Evaluation

23.3.1 Experiences with Labor Force Survey (LFS) data

In the first experiment with occupation coding in the LFS we obtained a p-value of 81 % and a q-value of 81 % (p and q as defined in chapter 20.2). A reconciliation of the 19 % not in accordance with the manual coding (corresponding to 111 disagreements) showed that the q-value for the PLEX dictionary was 87 % and the q-value for the SLEX dictionary was only 51 %. Thus, again the SLEX dictionary in this first version turned out to be inefficient.

The disagreements are distributed according to the following table:

Reason for disagreement	Type of dictionary	
	PLEX	SLEX
Coding error in automated system	49	45
Coding error in LFS	12	2
Coding error in both	1	1
Punching error	1	0
Σ	63	48

The errors due to SLEX are almost always real errors. The error-generating word strings are unsuitable and the best piece of advice is to delete them from the dictionary. Most of the errors due to PLEX occur because the computer cannot take advantage of the auxiliary information that sometimes can be utilized by the manual coder. Obviously some PLEX words are unsuitable but some can be used again with another code number. (Remember that this dictionary was simply translated from the census dictionary for the same variable.)

The most recent experiments indicate that with a PLEX constructed by computer we may for the time being obtain coding degrees around 70 % when coding LFS occupations. Nothing remarkable will happen if we carry out modest changes in this dictionary. If we combine PLEX with a very large alphabetic dictionary perhaps we can add another 5 % coded with an overall true error rate around 5 %. Unfortunately the error rate is a little bit too high to permit production use at the moment.

23.3.2 Experiences with coding of census occupations

Here we discuss the experiment with automated coding of census occupations in which the dictionary was constructed by the computer program.

The computerized construction resulted in 675 PLEX words and 523 SLEX words. The basic material was approximately 14 000 descriptions in size. The experimental material consisted of 3 796 verbal descriptions and the computer was able to code 2 632, or 69.3 %. 2 034 of these were coded by PLEX and the remaining 598 by SLEX. A comparison with manual coding of the same material showed 382 disagreements (14.5 %). 156 of these had been coded by PLEX and the remaining 226 by SLEX. Thus the disagreement rate for PLEX was 7.7 % and for SLEX 37.8 %. A closer examination of the PLEX disagreements reduced the error frequency to 5.6 %, since 42 of the 156 PLEX disagreements were in fact manual coding errors. As expected, with SLEX almost all the disagreements were due to unsuitable word strings in that dictionary: only 13 of the 226 disagreements were manual coding errors, which resulted in an error rate of 35.6 %. The overall error rate in this experiment was 12.4 %. The evaluation of the coding of occupation in this census showed that the error frequency was 13.5 %, which is somewhat higher. But we do know nothing about the error frequency in the material not coded by the computer. A seemingly strong argument is that this part is much more difficult to code than the rest of the

material. Unfortunately, we have not conducted any experiments to shed some light on this issue. A closer look at the material reveals that, as would be expected, the computer is unable to code complex (in fact long) and unusual descriptions. However, there are also many simple cases that the computer cannot handle - for instance, all those cases in which the coder uses the auxiliary information about industry which can be found on the schedule. All in all, though, it seems reasonable to believe that the part left uncoded by the computer is harder to code for a manual coder than the part coded.

When the errors due to the PLEX file were examined we found that approximately 50 PLEX descriptions were unsuitable and should be deleted. Some of them could be let stand if we merely changed the code number. When we examined the SLEX file we found that approximately 70 word strings should be deleted. However, with some manual modifications many of these could probably be reinserted with much improved results.

Let us finally describe the costs in this experiment (in US dollars):

Cost factors	Automated coding			Manual coding		
	Alt 1	Alt 2	Alt 3	Alt 1	Alt 2	Alt 3
Coding	80	80	80	930	700	560
Punching	300	300	300			
Recoding	300	215	170			
Σ	680	595	550	930	700	560

The cost figures for the manual recoding and the total manual coding are purely artificial and could merely be estimated, since manual coding of occupation was conducted together with coding of other variables. A manual recoding was in fact never carried out.

The different alternatives are based on different assumptions. Alternative 1 assumes a manual coding speed rate of 60 occupations per hour; 2 and 3 assume 80 and 100 respectively. Naturally, from an economic point of view, automated coding performs best when the manual coding operation is slow; with higher speeds the payoff vanishes. On the other hand, a higher coding degree and thus decreased cost for recoding could perhaps be obtained by changes in the dictionaries.

24 Swedish illustrations - goods

24.1 Experiments

In this section some experiments with automated coding of goods in the Household Expenditure Survey (HES) are described (see Lyberg (1978 a, b).

The goods in the HES have up to now been coded manually by means of a dictionary containing over 600 code numbers. Formally the code is built hierarchically but in practice it is just a list. The goods variable differs from occupation, for instance, in certain respects. Coding of goods demands none of the various kinds of judgments which are the big problem in coding occupation. Blood-pudding is almost always blood-pudding, while an architect could be assigned different occupation code numbers depending on the supplementary information available. An element of judgment can certainly be present in coding goods (for instance when the coder has to decide if an item of goods is fresh or frozen), but the normal case is not complicated. The observed error frequencies from earlier HES support this assumption. For instance, the coding error frequency was very low (less than 1 %) in the 1969 HES. Thus, the motive for experimenting with automated coding of this variable was not an alarming error frequency but was rather that, as in many other surveys, the coding procedure is costly and time-consuming. Our objective was to find out if automated coding could be used in the 1978 HES.

24.1.1 Materials and dictionaries

A sample of 26 000 records (= goods descriptions) was drawn from the 1969 HES. This sample was divided in four different parts. The largest one, consisting of 14 068 records, was used for computerized dictionary construction. The others, denoted E 1 (= 2 775 records), E 2 (= 4 823 records) and E 3 (= 4 713

records) respectively, were used as experimental materials. For each record the verbal description of the item of goods and its associated code number were punched on magnetic tapes. Generally the 1969 code number was assumed to be the correct one, in accordance with our prior information about error frequencies. The dictionary was constructed by means of the special computer program. The usual formatting technique was used and also certain uninformative word strings such as gallon, pound, yards, bottle, piece etc. were removed.

24.1.2 Results

The first dictionary construction created 686 PLEX words and 600 SLEX words. We used a frequency limit = 3, i.e. only those descriptions which occurred three or more times in the basic material were considered for the PLEX file.

The first automated coding gave the following results.

Results (trial 1, material E 2)

Number of descriptions	4 823	
Number of coded descriptions	3 305	(69 %)
Number coded in PLEX	2 840	
Number coded in SLEX	465	
Number not coded	1 518	
Number erroneously coded	246	(7 %)
in PLEX	32	
in SLEX	214	

Observe that we defined "erroneously coded" as "differing from the manual HES code number". Sometimes the latter was wrong. Reconciliation of the 246 differing cases gave the following results.

Error distribution

Source	Number of errors	%
HES 69	20	8
SLEX	209	85
PLEX	17	7
Total	246	100

Obviously the SLEX file is inefficient: it codes correctly in only 256 out of 465 possible cases. The PLEX file performs as well as the manual coders. In order to increase the coding degree we made a new dictionary with the same basic material but with the frequency limit decreased to 2. This dictionary created 1 140 PLEX words and 961 SLEX words. The same experimental material was used and the rerun gave the following results.

Results (trial 2, material E 2)

Number of descriptions	4 823	
Number of coded descriptions	3 594	(75 %)
Number coded in PLEX	3 045	
Number coded in SLEX	549	
Number not coded	1 229	
Number erroneously coded	375	(10 %)
in PLEX	40	
in SLEX	335	

The coding degree increased from 69 % to 75 % but the price we had to pay was an increased error frequency. The increase in coding degree might seem rather moderate considering that the size of the PLEX file almost doubled. This is explained by the fact that the distribution over categories is far from rectangular: a small number of categories is enough to code a large number of records. Thus additional and somewhat unusual descriptions do not have any remarkable impact on the outcome.

The analysis of trials 1 and 2 pointed to the fact that the state of food products (fresh, frozen etc.) accounted for 13 % of the errors in the automated coding. About 8 % of the uncoded descriptions would have been coded if separation between states had not been necessary. The analysis gave hints concerning certain changes in the dictionary, and some words were added to the PLEX file and some were removed. At this time we were already considering the possibility of rejecting the SLEX file since it was inefficient. As can be seen from the results above, coding using only the PLEX file would give the following.

Results (trial 3 = trial 2, material E 2)

Number of descriptions	4 823
Number of coded descriptions	3 045 (63 %)
Number of coded in PLEX	3 045
Number of erroneously coded	40 (1 %)

We have reason to believe that only half of the errors are real ones. With this coding we obtain a quality at least as good as the manual coding and a coding degree of 63 %.

Now a new independent experimental material consisting of 4 713 records (E 3) was used with a dictionary slightly larger than that used in the previous experiment. The following results were found.

Results (trial 4, material E 3)

Number of descriptions	4 713
Number of coded descriptions	3 800 (81 %)
Number coded in PLEX	3 175
Number coded in SLEX	625
Number not coded	913
Number erroneously coded	271 (7 %)
in PLEX	36
in SLEX	235

This time the quality and the coding degree increased a little. A coding with only the PLEX file would have resulted in a 67 % coding degree and a 1 % "error" rate.

Now the last of the experimental materials consisting of 2 775 records (E 1) was run.

Results (trial 5, material E 1)

Number of descriptions	2 775	
Number of coded descriptions	2 175	(78 %)
Number coded in PLEX	1 859	
Number coded in SLEX	316	
Number not coded	600	
Number erroneously coded	175	(8 %)
in PLEX	33	
in SLEX	142	

This time the coding degree as well as the quality decreased somewhat. Coding by means of the PLEX file would have given a 67 % coding degree only and a 1 % "error" rate, the same as in the previous trial.

We now used the results from the trials with E 1 and E 2 to make a list of removals and additions to the PLEX dictionary, and a new modified PLEX file was created. Then E 3 was run once more. This enabled us to control the effect of the improved dictionary.

Results (trial 6, material E 3)

Number of descriptions	4 713	
Number of coded descriptions	3 680	(78 %)
Number coded in PLEX	3 192	
Number coded in SLEX	488	
Number not coded	1 033	
Number erroneously coded	168	(4.5 %)
in PLEX	23	
in SLEX	145	

The coding degree decreased from 81 % to 78 %. On the other hand, the quality increased: the error rate decreased from 8 % to 4.5 %. A coding using only the new PLEX file would have resulted in a 68 % coding degree and an error rate of .6 %.

The series of experiments was terminated by a study showing how coding degree and quality are influenced by the size of the dictionary. Let us first turn back to the results obtained in trial 5. In that trial a dictionary based on 14 000 descriptions was used with a frequency value = 2. Now we wanted to see what would happen with the coding if the basic material was extended, first with material E 2 and subsequently with E 2 and E 3. Thus the special program made two different dictionaries to be compared with the one already available. The sizes of the dictionaries and the results of the associated coding are shown below.

Results (trial 7, material E 1)

File	Number of records in the dictionary		
	Original (L)	L + E 2	L + E 2 + E 3
PLEX	1 140	1 378	1 610
SLEX	961	1 157	1 314
Size of basic material	14 068	18 891	23 604

	Dictionary		
	L	L + E2	L + E 2 + E 3
Number of descriptions	2 775	2 775	2 775
Number of coded descriptions	2 175	2 226	2 217
Number coded in PLEX	1 859	1 905	1 935
Number coded in SLEX	316	321	282
Number not coded	600	549	558
Number erroneously coded	175	179	160
in PLEX	33	33	34
in SLEX	142	146	126

The coding degree increased at first by 2 percentage points to an 80 % level. However, using the largest dictionary is associated with a small decrease in coding degree. We hope that this paradox is explained by the fact that the SLEX file is used more seldom here. The error frequency decreases somewhat when the dictionary size increases. If the PLEX file alone is used, the coding degree increases from 67 % to 70 % when going from the smallest to the largest dictionary. Again, the results illustrate the fact that a relatively small number of dictionary descriptions is sufficient to code a large part of the records. The most common descriptions are already contained in a moderate-sized dictionary. Beyond that point additional descriptions tend to be rather unusual.

24.2 Applications

The experiments with coding of goods in 1969 HES material were so encouraging that it was decided to try it in regular production. In this section some results and experiences from this regular production are presented.

In the 1978 HES a random sample of approximately 13 800 households was selected. Of these, approximately 5 900 households were supposed to keep a complete diary (CD) of all goods purchased during a two-week period. The rest of the sample was supposed to keep a simplified diary (SD) of goods purchased during a four-week period. In SD, only those goods that cost more than 50 kronor (approximately 12 US dollars) should be considered, and no food products needed to be noted, whatever their cost.

The survey design allowed continuous delivery of diaries from the respondents. Thus the material could be processed in cycles, which might be advantageous in a system with automated coding. The experimental results given in section 24.1 led to the decision that automated coding should be carried out in the survey,

at least to some extent. Naturally we were not at all convinced that our system should work in a production environment. Thus, it was decided that, to start with, the coding should be carried out by two parallel systems, one manual and one automated. It was also decided that after two months the performance of the systems should be evaluated, and afterwards the preferred system would be used during the remaining ten months of processing.

24.2.1 The automated system

After arrival at Statistics Sweden the diary books were clerically preedited in order to facilitate punching. After that the precoded items, id's, and so forth were keypunched together with the alphabetic descriptions of the goods. Then the descriptions were matched against a computer-stored dictionary and the matching resulted in coded and uncoded descriptions. Uncoded descriptions were listed alphabetically on an optical character recognition form and code numbers were assigned directly on it. Some of the uncoded descriptions were added to the dictionary in the updating process: the forms were read and the code numbers were merged with those already assigned by the computer, so that no further keypunching was necessary.

24.2.2 Dictionary construction

The programs for dictionary updating, coding and production statistics were written in FORTRAN and ASSEMBLER. The AUTOCOD package could not be used directly since for various reasons the computer processing of this survey was not carried out by Statistics Sweden. However, the programs were constructed in the same way as AUTOCOD.

The dictionary construction was step-wise. Relatively extensive efforts were made in creating an initial dictionary. After that continuous revisions were made prior to almost all cycles.

The initial dictionary was based on the dictionary used in the experiments with the 1969 HES material together with a list of all descriptions in the experimental material (consisting of 26 000 items). Each unique description was coded by HES experts.

The construction involved a lot of manual work since the pattern of descriptions had changed during the nine years that had passed since the last HES. Only a PLEX was constructed, with an unequivocal rate of T_1 = 100 %. No relatively long and detailed descriptions of the type "oil for the car" or "acrylic sweater" were permitted in PLEX. In addition, no descriptions mixing good and trademarks were permitted: for instance, "Goodyear tires" or "Wilson tennis balls" were not admissible.

This initial dictionary consisted of 1 459 descriptions.

24.2.3 Results

During the period from March 15 to November 18, 1978, 19 cycles were run. During this period 14 different versions of PLEX were used; thus only a few cycles were coded with identical dictionaries. As this is being written the survey is still going on, but in May 1978 it was decided that the coding from then on should be automated. In Table 24.1 below the coding degrees for the 19 cycles are given.

Table 24.1 Coding degree for the first 19 cycles in the 1978 HES.

Dictionary version	Cycle	Coding degree (%)
1	1	56
2	2	63
3	3	67
4	4	66
5	5	68
5	6	68
5	7	63
6	8	64
7	9	53
8	10	69
8	11	66
9	12	61
10	13	68
11	14	58
12	15	72
12	16	73
13	17	39
13	18	70
14	19	65

The coding degree over all cycles was 66 %. As can be seen from the table, the coding degree decreases now and then: for cycle 17 it is less than 40 % while neighboring cycles have high values. At first we were taken off guard. However, a closer look revealed that the proportion of simplified diaries could vary substantially. Not surprisingly, it was discovered that cycle 16 consisted of CD's only while cycle 17 consisted of SD's only. In the other cycles the different types were mixed in various proportions. By means of a least-square estimator we were able to estimate the coding degree for CD as 70 % and the coding degree for SD as 38 %.

24.2.4 Updating the dictionary

The dictionary was modified prior to most of the cycle runs. Generally the additions have outnumbered the removals, as shown in Table 24.2.

Table 24.2 Dictionary sizes in the 1978 HES.

Dictionary version	Number of descriptions	Used in no. of cycles
1	1459	1
2	1554	1
3	1760	1
4	2228	1
5	2464	3
6	1632	1
7	1990	1
8	2451	2
9	2866	1
10	3065	1
11	3613	1
12	3752	2
13	3832	2
14	4011	?

An extensive reduction of the dictionary size took place in the creation of version 6, mainly in order to reduce costs. It is true that the computer processing costs do not depend greatly on the size of the dictionary, but during this first application it was felt necessary to list the different dictionary versions in several copies in order to provide each HES staff member with his own copy. The cost of these listings was considered non-trivial. As can be seen in table 24.1, this reduction had no effect on the coding degree. During the whole period the dictionary size has more than doubled but the coding degree has changed little. Prior to the reduction it was investigated how the 2464 descriptions had been used during the previous cycles. 832 descriptions had been used in fewer than three households, and 312 descriptions had never been used. During these first cycles 15 109 unique descriptions concerning 79 311 purchases appeared.

All new uncoded descriptions were gathered in a special file. As soon as a new description occurred in three households it was included in the dictionary, provided it could be unequivocally coded to a specific category.

The cumulative number of unique descriptions follows the cumulative number of purchases almost linearly, at least for the first 14 cycles from which we have such data. Up to the 14th cycle 205 515 purchase descriptions had been keypunched to be forwarded to automated coding. These purchase descriptions were associated with 35 994 unique descriptions. Thus, although it may sound trivial, we can conclude that the number of such descriptions increases all the time and that we have not yet seen an end of it. A more detailed description of the production is found in Lyberg (1978 b).

24.3 Evaluation

24.3.1 General experiences

Let us return for a moment to the results given in section 24.1 concerning the coding of goods. Consider the Table 24.3.

Table 24.3 Some results of experiments with automated coding of goods.

Experimental material	Coding degree p(%)	Quality (agreement rate) q(%)	Dictionary sizes
A	69	93	686 PLEX + 600 SLEX, frequency (mincount) 3
A	75	90	1140 PLEX + 961 SLEX, frequency 2
A	63	99	1140 PLEX, frequency 2
B	81	93	1140 PLEX + 961 SLEX, frequency 2
B	67	99	1140 PLEX, frequency 2
B	78	96	Modified by independent materials
C	78	92	1140 PLEX + 961 SLEX, frequency 2
C	67	99	1140 PLEX, frequency 2
C	80	92	1378 PLEX + 1157 SLEX, frequency 2
C	80	93	1610 PLEX + 1314 SLEX, frequency 2

The table clearly shows that we have great difficulties in increasing the q-value as long as we make use of both PLEX and SLEX. When we only use PLEX, q reaches the 99 % level. The price paid is the lower coding degree: the value of p goes down by 11-14 percentage points in the three materials when the SLEX dictionary is dropped. When we considered automated coding in the 1978 HES we started to look at the q-values. We had some experiments from the last HES conducted in 1969 showing that the manual coding quality was very good: the frequency of erroneous coding was probably somewhere around one percent.

Considering the other specific sources of non-sampling errors occuring in a HES, such as non-response and reporting errors, we were not willing to trade this high-quality coding process just for money. Thus we decided to aim at a q-value around 99 %. The consequence was that the inefficient SLEX file had to be dropped. Now the remaining problem was to investigate whether a p-value around 67 % could justify the use of auto-

mation. Preliminary calculations showed that it could and it was decided that the 1978 HES should be coded by a PLEX, at least to start with (remember the parallel manual system).

Now let us examine the coding errors in some of the previously described experiments. Starting with the experiment with material E 3 in the HES, it was found that 271 of the 3 175 coded descriptions of goods were not in accordance with the code number assigned manually. 235 of the cases of disagreement had been coded by the SLEX dictionary and the remaining 36 by the PLEX dictionary. A closer look revealed that half of the PLEX disagreements were due to errors in the manual coding. The rest was due to errors in judging the state (for instance, canned meat was judged to be fresh). We were not able to find any unsuitable dictionary words in PLEX. However, when studying the SLEX disagreements we found that only 6 of the 235 disagreements were due to manual coding errors and 36 were due to errors in judging the state. The rest, 192 cases, turned out to be result from the presence of unsuitable dictionary word strings. Thus these unsuitable descriptions should be deleted from our 961-description SLEX dictionary.

24.3.2 Experiences with the production of the 1978 HES

The experiments with coding of goods indicated an expected coding degree of around 67 %. We could not be quite sure of obtaining this rate in the 1978 survey - far from it - since the experiments were conducted with non-random samples from nine-year-old material. Furthermore, automated coding had never before been applied in Statistics Sweden. Large-scale keypunching of alphabetic descriptions was an entirely new production situation and we did not know how computer routines would work together with manual ones. In addition, the cost estimates had been rather artificial.

However, so far we have an overall coding degree of 66 %. The key operators find it less boring to punch verbal information for a change. All in all it has been a smooth operation. The cost calculations point to the fact that automated coding is 2-5 % cheaper than the manual system. Besides, the system will provide some further advantages. Since all descriptions are keypunched the primary material is documented better than when merely the code number is keyed. Thus it is possible to give more detailed descriptions of the goods contained in the groups for which estimates are provided. Furthermore, since the dictionary manages to code all straightforward descriptions the remaining manual coding becomes more interesting to the coder.

The computer coding itself is cheap. For instance, in cycle 12, 7166 purchases were entered into the computer. This run, with a resulting coding degree of 61 %, cost 226 kronor (approximately 50 US dollars). In cycle 18, the coding of 34 070 purchases with 70 % coding degree cost 327 kronor (74 dollars). The cost of updating dictionary number 13, for instance, was 366 kronor (83 dollars). We have noted that the manual preparation of runs is the expensive part in this automated system. Thus if the number of dictionary modifications and the number of cycles are reduced this will also produce considerable cost reduction.

The coding degrees in SD's and CD's differ substantially. We did not realize that the large cycle variations in overall coding degree originated from that fact until several cycles had been processed. If we had known then what we know today, then perhaps the purchases gathered via SD's might have been coded manually from the start. In fact, we did not suspect anything until the processing of cycle 9, and at that time we were already dealing with dictionary version number 8. Fortunately the low SD coding degree affects the overall coding degree very little since the SD purchases are far outnumbered by the CD purchases.

We have already noted that the cumulative number of unique descriptions is almost linearly related to the cumulative number of purchases entered into the computer. Let us look at Table 24.4.

Table 24.4 Number of unique descriptions in the first 1978 HES cycles.

Cycle	No. of purchases	No. of new unique descriptions	Cumulative no. of unique descriptions
1, 2	17 424	4 648	4 648
3, 4	40 239	6 411	11 059
5, 6, 7	26 463	4 050	15 109
8	14 992	2 222	17 336
9	12 894	3 039	20 375
10, 11	29 280	4 555	24 930
12	7 166	1 290	26 220
13	34 284	4 957	31 177
14	22 773	4 813	35 994

The table shows that for each new batch of purchases a large number of new unique descriptions is generated. Thus after cycle 14 we had encountered about 36 000 unique descriptions. Of these only about 3 600, or 10 %, were present in the dictionary used in that cycle. Furthermore, we know that of these 3 600 only a fraction is used very often, and some of them are used almost not at all.

It does not seem worth the effort to make extensive dictionary revisions after a specific point. Quite soon a rather stable coding degree is obtained which cannot be substantially altered without changing the dictionary construction principle. We can note that with the third version we had already obtained a coding degree of 67 %. Despite much work and repetitive modifications after that point, we have obtained at best 73 %.

With a SLEX it is possible to increase the coding degree by, say, up to 13 percentage points. Such an increase would be accompanied by an increased error rate from the present PLEX rate, which is less than .5 %, to a value somewhere in the interval .5 - 10 %.

25 Swedish illustrations - book loans

25.1 Experiments

The Swedish Author's Fund makes disbursements to authors in proportion to the popularity of their books among borrowers at public libraries. This bonus is based on sample data from different libraries in Sweden and it is distributed once a year. The survey is carried out by Statistics Sweden on a commission basis.

Earlier the loans were coded manually on a computer list containing alphabetic keypunched authors and book titles. This situation is unique with respect to automated coding: we already have punched descriptions of the variable under study; in such a situation it is easy for an automated system to compete with the manual. Even a rather modest coding degree makes the automated system profitable, since the punching is free of charge. The only requirement is that the computer cost should be less than the manual coding cost on a record-by-record basis.

Only a PLEX dictionary with an unequivocal rate T_1 = 100 % was considered since each error could have a substantial effect on the bonuses distributed. Anayway, at the time it was no use arguing over this matter with the survey sponsors.

Prior to the 1978 survey an experiment was conducted. An old file comprising about 14 000 records (author and book title) was available. 10 000 records were used as a basic file for dictionary construction by means of our program package. The resulting dictionary consisted of 6 900 word strings. The remaining 4 000 records were used as a test file. The resulting coding degree was 47 %.

25.2 Applications

The experiment was somewhat incestuous since the loans in the two files were sampled from the same set of counties. This fact, together perhaps with other unknown factors, explains the lower coding degree of 33 % in the production of the 1978 survey. However, the favorable conditions still made the new system profitable by a broad margin.

25.3 Evaluations

No special evaluation studies have been conducted due to the strong requirement on T_1.

26 Swedish illustrations - medicine

The Swedish Pharmaceutical Company conducts continuous surveys of medicine consumption in Sweden. The sampling units are the prescriptions. Each type of medicine is defined by a code number (unfortunately, this number is not assigned by the physician). In order to reduce processing time, a system which transfers data from the prescriptions to punch cards without manual coding has been developed.

For each prescription and medicine the following information is punched and used as input:

i) The three initial letters in the name of the medicine
ii) The number of packets
iii) The price

This information is matched against a dictionary of medicines and associated code numbers (combined with decision tables when prices and number of packets are used). The system coding degree is around 90 % with a 1 % error rate.

The system is described in Grantzell et al. (1975).

EPILOGUE

The background: increasing use of administrative records and development of computers

The future of coding control, like that of other fields, is hard to predict. However, we know a few things for sure. We know that administrative records will be increasingly used for statistical purposes. As pointed out in Cartwright and Armknecht (1979), the primary distinction between administrative and statistical records is the degree to which the statistician is in control of the design and collection of records. In recent years a more intensive use of existing administrative records as sources of data for statistical purposes has been called for, at least in the USA and Sweden. There are many reasons for this demand. Among other things, the very existence of administrative records calls for an extended use for statistical purposes. Furthermore the respondent burden, heavy as it already is, must not be allowed to increase.

There are reasons to believe that in some cases respondents are more willing to report to an administrative system than respond to a statistical survey. In those cases it seems obvious that a statistical system should coordinate its data collection with a suitable administrative system. Such coordination presents problems of its own, e.g. in differences as to definitions, data collection procedures, etc. Furthermore, the legal issues are far from resolved.

An efficient link between administrative records and statistics production implies the necessity of uniform coding. Uniform coding means uniform codes; i.e., for the variable under study a specific verbal description should be assigned the same code number irrespective of the system. As we have seen, this is not always the case even within statistics production. Uniform coding is the necessary basis for effective coding control and is therefore one of the most urgent tasks.

Coding within administrative systems is of course very sensitive to errors since each error can have important effects on the individuals for whom the system is designed. The coding control must therefore be tighter than in statistical systems, where the errors sometimes tend to cancel out. However, special care must be taken in using administrative systems in statistics production. Suppose that data in an administrative system are used to construct sampling frames for statistics production. Then, again, each individual error might constitute a serious flaw.

Thus, it seems obvious that one can expect in the future more intensive use of efficient control methods such as independent verification.

The development of computers has been important over the years. We have seen, for instance, that the replacement of manual matching of code numbers by computer matching facilitates things a great deal. Recent studies show that data processing can be made more efficient by letting a single clerk carry out several processing tasks more or less simultaneously. For instance, in Rustemayer (1977) it is shown how interviewing can be computer-assisted by means of a system in which telephone interviewing is carried out at a computer terminal. No wild fantasy is needed to imagine that such a procedure can be extended to cover such operations as editing, coding and punching as well. Concentration of several operations, including coding, in a centralized facility makes things a lot easier. Coding instructions can be displayed on terminals and quick changes in codes and instructions can be entered on terminals. Verification and feedback can be conducted more easily in a computerized environment. Coding in such an environment should be a cheaper and more effective operation.

The future of automated coding

Obviously the success of automated coding is a function of language complexity. It seems that the Swedish language is more forgiving than English in this respect. Our experiments have shown that automated coding might be a possible option when designing the coding operation. Most of our experiments deal with methods that are rather unsophisticated. Our efforts with secondary dictionaries, distance rules and correlation coefficients have not been very successful so far with respect to quality. On the other hand, these efforts have not been especially extensive. We think, however, that using a SLEX of moderate size can be profitable both economically and in terms of quality. In the US Bureau of the Census automated coding is used for coding geographic locations. Currently they are striving for an automated system to handle the variables occupation and industry, but no system has been found ready for implementation.

In Statistics Sweden we were able to use automated coding in coding goods in the 1978 Household Expenditure Survey. However, to make the system more profitable we should like the coding degree to jump from the current 66 % to, say, 80 or 85 %. This could be done if the code could be changed in some sections. For instance, gasoline in various forms cannot be handled by the computer since this item is assigned different code numbers depending on what kind of vehicle it is intended for. The large part of the gas is used for cars, but because of obligations towards the data users concerning providing special gas estimates it was impossible to merge these categories. And we can assure you that there are many gas purchases in the diaries. The same goes for a number of other goods. The coding degree can also be improved by storing prices and quantities in the dictionaries, and by using a small but effective SLEX the goal of 80 - 85 %

could be achieved in the next HES. This "gasoline" problem is general. Perhaps it is not too preposterous to make changes in the codes in order to obtain a less costly coding. That option should certainly be considered more often in times of scarce financial resources.

Our experiments with industry and occupation show that occupation is much easier to code than industry. It may well be possible to use automated coding in the Labor Force Survey. Currently we are very close to the payoff point in that survey. Automated coding of census occupations is also possible and is considered for the 1980 Swedish Census of Population. Our experiments with industry have been very limited since this variable is semantically much more complex than occupation. (Curiously, overseas it is the other way around.) Our strategy has been to put the easier variables to a test first, and then, if possible, proceed to the more difficult ones.

REFERENCES

Andersson, R. (1974): Skattning av totaler med utnyttjande av supplementär information. Metodinformation nr 74:14, Statistics Sweden (In Swedish).

Andersson, R. (1977): Kodningskvaliteten i FoB 75 - ekonomisk aktivitet. Memo, Statistics Sweden (In Swedish).

Bachi, R., Baron, R., and Nathan, G. (1967): Methods of Record-Linkage and Applications in Israel. Bulletin of International Statistical Institute.

Bailar, B.A. and Dalenius, T. (1969): Estimating the Response Variance Components of the US Bureau of the Census' Survey Model. Sankhyā, Series B, vol 31, Parts 3 & 4, pp 341-360.

Barnett, V. (1973): Bayesian and Decision Theoretic Methods Applied to Industrial Problems. The Statistician, pp 199-226.

Bauman, K.E. and Chase, C.L. (1974): Interviewers as Coders of Occupation. Public Opinion Quarterly, Spring, pp 107-112.

Bennett, E.M., Alpert, R., and Goldstein, A.C. (1954): Communications Through Limited-Response Questioning. Public Opinion Quarterly, pp 303-308.

Berg, S. (1974): Personal communication.

Biemer, P.P. (n.d.): An Improved Procedure for Estimating the Components of Response Variance in Complex Surveys. Unpublished memo.

Biggs, J.F. (1970): Quality Control Applied to Clerical Coding. Paper presented at the American Society for Quality Control, Arlington, Virginia.

Blair, C.R. (1960): A Program for Correcting Spelling Errors. Information and Control, Vol 3, pp 60-67.

Boston, G.F.P. (1977 a): Quality Control of Occupation Coding. Unpublished draft, Office of Population Censuses and Surveys, U K.

Boston, G.F.P. (1977 b): Machine Coding of Occupation. Unpublished memo, Office of Population Censuses and Surveys, U K.

Boston, G.F.P. (1978): Computer Coding of Occupations. Paper MG 67, Office of Population Censuses and Surveys, U K.

Bostrom, C. (1975): Specifications for the Simple Circuit Generator (also known as the Corbett Algorithm or the Minimal Circuit Machine). Memo, US Bureau of the Census.

Brodda, B. and Karlgren, H. (1968): Trade-Mark Similarity Check by Computer. Språkförlaget SKRIPTOR.

Brodda, B. and Karlgren, H. (1969): Mechanized Trade-Mark Examination. Språkförlaget SKRIPTOR.

Bross (1954): Misclassification in 2 x 2 Tables. Biometrics, pp 478-486.

Bäcklund, S. (1978): Automatisk kodning. Beskrivning av programvara och programvaruhantering. Memo, Statistics Sweden (In Swedish).

Cartwright, D.W. and Armknecht, P.A. (1979): Statistical Uses of Administrative Records. Paper prepared for the American Statistical Association Annual Meetings, August, 1979.

Casady, R.J. (1966): A Model for Analysis of Reinterview Data in the NCHS Health Interview Survey, Memo, US Department of Health, Education and Welfare.

Christianson, A. (1972): Felproblematiken utanför surveyområdet -medicinsk diagnostik. Rapport nr 43, Forskningsprojektet FEL I UNDERSÖKNINGAR, Stockholms universitet (In Swedish).

Cook, W.H. (1959): A Point System of Quality Control. In: American Society for Quality Control. Science in Management. Proceedings of the 1959 conference of the Administrative Applications Division. Washington, DC, pp 207-209.

Corbett, J.P. (1972): Encoding from Free Word Descriptions. Memo, US Bureau of the Census.

Cornew, R.W. (1968): A Statistical Method of Spelling Correction. Information and Control, vol 12, pp 79-93.

Coulter, R.W. (1977): An Application of a Theory for Record Linkage. Memo, US Department of Agriculture.

Crittenden, K.S. and Hill, R.J. (1971): Coding Reliability and Validity of Interview Data. American Sociological Review, vol 36, no 6, pp 1073-80.

Dalenius, T. (1962): Recent Advances in Sample Survey Theory and Methods, Annals of Mathematical Statistics, vol. 33, pp 325-349.

Dalenius, T. (1974): Ends and Means of Total Survey Design. Forskningsprojektet FEL I UNDERSÖKNINGAR, Stockholms universitet, Statistiska Institutionen.

Dalenius, T. and Frank, O. (1968): Control of Classification. Review of the International Statistical Institute, vol 36:3, pp 279-295.

Dalenius, T. and Lyberg, L. (n.d.): An Experimental Comparison of Dependent and Independent Verification of Coding. Memo from Tore Dalenius to Leon Pritzker.

Danmarks statistik (1976): Erhvervsstatus - og stillingskodesystemet. Memo, Danmarks Statistik (In Danish).

Damerau, F. (1964): A Technique for Computer Detection and Correction of Spelling Errors. Communications of the ACM, pp 171-176.

Davidson, L. (1962): Retrieval of Misspelled Names in an Airlines Passenger Record System. Communications of the ACM, pp 169-171.

Davidson, L. (1972): Personal communication.

Dixon, N.R. and Tappert, C.C. (1971): Strategic Compromise and Modeling in Automatic Recognition of Continuous Speech: A Hierarchical Approach. Journal of Cybernetics, pp 67-79.

Dolby, J.L. (1977): On the Notions of Ambiguity and Information Loss. Information Processing & Management, pp 69-77.

Dodge, H.F. and Romig, H.G. (1944): Sampling Inspection Tables. Wiley.

Du Bois, Jr., N.S.D. (1969): A Solution to the Problem of Linking Multivariate Documents. Journal of the American Statistical Association, pp 163-174.

Eckhardt, K.W. and Wenger, D.E. (1975): Respondent Coding of Occupation. Public Opinion Quarterly, Summer, pp 246-254.

Estabrook, G.F. and Rogers, D.J. (1966): A General Method of Taxonomic Description for a Computed Similarity Measure. Bio Science, pp 789-793.

Fasteau, H.H., Ingram, J.J., and Mills, R.H. (1962): Study of the Reliability of Coding of Census Returns. Proceedings of Social Statistics Section, American Statistical Association, pp 104-115.

Fasteau, H.H., Ingram, J.J. and Minton, G. (1964): Control of Quality of Coding in the 1960 Censuses. Journal of the American Statistical Association, vol 59, no 305, pp 120-132.

Fellegi, I.P. (1964): Response Variance and Its Estimation. Journal of the American Statistical Association, vol 59, pp 1016-1041.

Felme, S. (1978): Konstruktion av statistiska koder. Memo, Statistics Sweden (In Swedish).

Felme, S. (1980): Kodkonstruktion. Metodinformation nr 80:1, Statistics Sweden (In Swedish).

Felme, S., Lyberg, L., and Olsson, L. (1977): Kvalitetsskydd av data. Liber förlag (In Swedish).

Frisbie, B. and Sudman, S. (1968): The Use of Computers in Coding Free Responses. Public Opinion Quarterly, pp 216-232.

Funkhouser, G.R. and Parker, E.B. (1968): Analyzing Coding Reliability: The Random-Systematic-Error Coefficient. Public Opinion Quarterly, pp 122-128.

Giesecke, G.L. (1973): A Comparison of Computerized Techniques for Recognizing Spanish Names. US Office Secretary of Defense, Report no. MA 73-2.

Grantzell, A., Kristoferson, K., and Wessling, A. (1975): Kontinuerlig kartläggning av läkemedelskonsumtionen i Sverige. Svensk Farmaceutisk Tidskrift, 1975: 9, pp 483-491 (In Swedish).

Guild, R.D. (1976): Effective Sampling Plans. American Society for Quality Control, Technical Conference Transactions, pp 285.

Gustavsson, R. and Karlsson, J-E. (1977): Beskrivning av rutiner för automatisk kodning (Version 3). Memo, Statistics Sweden (In Swedish).

Gustavsson, R. and Karlsson, J-E. (1978): Automatisk kodning. Memo, Statistics Sweden (In Swedish).

Hald, A. (1960): The Compound Hypergeometric Distribution and a System of Single Sampling Inspection Plans Based on Prior Distributions and Costs. Technometrics, vol. 2, no 3, pp 275-340.

Hald, A. (1963): Efficiency of Sampling Inspection Plans for Attributes. Bulletin of International Statistical Institute, 40, pp 681-698.

Hald, A. (1968): Bayesian Single Sampling Attribute Plans for Continuous Prior Distributions. Technometrics, pp 667-683.

Hansen, M.H., Hurwitz, W.N., and Bershad, M.A. (1961): Measurement Errors in Censuses and Surveys. Bulletin of the International Statistical Institute, vol 38:2, pp 359-374.

Hansen, M.H., Hurwitz, W.N. and Pritzker, L. (1964): The Estimation and Interpretation of Gross Differences and the Simple Response Variance. In 'Contributions to Statistics', presented to Professor P.C. Mahalanobis on the occasion of his 70th birthday, pp 111-136.

Harris, K. (1974): Analysis of the Independent Three-Way Verification System in Mortality Medical Coding. Memo, US Department of Health, Education and Welfare.

Hartley, H.O. and Rao, J.N.K. (1978): Estimation of Nonsampling Variance Components in Sample Surveys. In Namboodiri, N.K. (Ed.): Survey Sampling and Measurement, pp 35-43, Academic Press.

Harvig, H. (1973 a): Kontrollkodning av dödsbevis. Memo, Statistics Sweden (In Swedish).

Harvig, H. (1973 b): Kontrollkodningsexperiment på blanketter för inskrivningsuppgifter till högre studier. Memo, Statistics Sweden (In Swedish).

Harvig, H. (1975 a): Kontrollkodning av sjuksköterskestatistiken. Memo, Statistics Sweden (In Swedish).

Harvig, H. (1975 b): Kontrollkodning av yrkesvariabeln i "Motiv för vuxenstudier vt 74". Memo, Statistics Sweden (In Swedish).

Harvig, H. (1975 c): Kontrollkodning av tre variabler i ULFen. Memo, Statistics Sweden (In Swedish).

Hendricks, W.A. (1964): Estimation of the Probability that an Observation will Fall into a Specific Class. Journal of the American Statistical Association, pp 225-232.

Hinrichs, J.R., and Gatewood, R.D. (1967): Differences in Opinion Survey Response Patterns as a Function of Different Methods of Survey Administration.(No other data available.)

Hochstim, J.R. (1967): A Critical Comparison of Three Strategies of Collections Data from Households. Journal of the American Statistical Association, vol 62, pp 976-989.

Hyvärinen, L. (1971): Personal communication.

IBM (1969): Identification Techniques. IBM Data Processing Techniques. IBM C 20-1707-0.

Iker, H.P. and Harway, N.I. (1965): A Computer Approach Towards the Analysis of Content. Computers in Behavioral Science, pp 173-182.

Israel, R.A., Templeton, M.C., and Evans, M.C. (1972): New Approaches to Coding and Analyzing Mortality Data. American Statistical Association, Proceedings of Social Statistics Section, pp 20-24.

Jabine, T.B. (1970): Quality Control of Coding of the 1970 Decennial Census Sample Questionnaires. Memo, US Bureau of the Census.

Jabine, T.B. and Tepping, B.J. (1973): Controlling the Quality of Occupation and Industry Data. Invited paper presented at the International Statistical Institute meeting in Austria, 1973.

Jackson, M. (1967): Mnemonics. Datamation, pp 26-28.

Jansson, Å. (1978): Fel vid kodning av brott. Kvalitetskontroll av brottsstatistiken. Promemorior från SCB 1978:1 (In Swedish).

Jaro, M.A. (1972): UNIMATCH-A Computer System for Generalized Record Linkage under Conditions of Uncertainty. AFIPS-Conference Proceedings, Vol. 40, AFIPS Press.

Juran, J.M. (1968): Operator Errors-Time for A New Look. Quality Progress, February, pp 9-11.

Karlsson, J.-E. (1978): Hjälpmedel vid automatisk kodning. ADB-information, nr. P O 9. 025, Statistics Sweden (In Swedish).

Kay, M. and Martins, G.R. (1970): The Mind System: The Morphological - Analysis Program. The RAND Corporation, California, USA.

Kellerman, E. (1972): An Adaptive Query System. Paper presented at the APL Users Conference 1972.

Knaus, R. (n.d.): Syntactically Based Classification from Natural Language Responses. Memo, US Bureau of the Census.

Knaus, R. (1978 a): Inference by Semantic Pattern Matching in Industry Classification. Memo, US Bureau of the Census.

Knaus, R. (1978 b): Automated Industry Coding - an Artificial Intelligence Approach. Memo, US Bureau of the Census.

Knaus, R. (1979): A Similarity Measure on Semantic Network Nodes. Paper presented at the Classification Society Annual Meeting, Gainesville, Florida, 1979.

Krause, P. (1971): History of Coding Verification, 1970 Census. Memo, US Bureau of the Census.

Krohn, M., Waldo, G.P., and Chiricos, T.G. (1975): Self-reported Delinquency: A Comparison of Structured Interviews and Self-administered Checklists. The Journal of Criminal Law and Criminology.

Lakatos, E. (1977 a): Automated I & O Coding. Memo, US Bureau of the Census.

Lakatos, E. (1977 b): Computerized Coding of Free Verbal Responses. Memo, US Bureau of the Census.

Lavin, M. (1946): Inspection Efficiency and Sampling Inspection Plans. Journal of the American Statistical Association, pp 432-438.

Linebarger, J.S., Jablin, C., and Davie, W. C. (1976): Dependent versus Independent Verification. Memo, US Bureau of the Census.

Lyberg, I., Haglund, T., Lyberg L., and Wretman, J. (1978): The Allocation of Resources to Production and Control Steps in the Statistical Process. Statistisk Tidskrift, no 4, pp 253-276.

Lyberg, L. (n.d.): Beroende och oberoende kontroll av kodning. Rapport nr 4, Forskningsprojektet FEL I UNDERSÖKNINGAR, Stockholms universitet, Stockholm (In Swedish).

Lyberg, L. (1969): On the Formation of Coding Teams in the Case of Independent Verification Under Cost Considerations. Forskningsprojektet FEL I UNDERSÖKNINGAR, Stockholms universitet, Stockholm, rapport nr 18.

Lyberg, L. (1970): Resultat av prov med oberoende kontrollkodning. Memo, Statistics Sweden (In Swedish).

Lyberg, L. (1972 a): Den s k snabbevalveringen av kodningen i FoB 70. Memo, Statistics Sweden (In Swedish).

Lyberg, L. (1972 b): A Note on Automatic Coding. Statistisk Tidskrift, nr 2, pp 137-142.

Lyberg, L. (1974 a): Automatic Classification of Industry and Occupation - Some Experiments. Memo, Statistics Sweden.

Lyberg, L. (1974 b): Kodningskvaliteten i FoB 70 - ekonomisk aktivitet. Memo, Statistics Sweden (In Swedish).

Lyberg, L. (1974 c): En studie av några s k 1-1-fall. Memo, Statistics Sweden (In Swedish).

Lyberg, L. (1976): Automatisk kodning. Memo, Statistics Sweden (In Swedish).

Lyberg, L. (1977 a): Coding of Verbal Information. Proceedings of the Social Statistics Section, American Statistical Association, Part I, pp 354-359.

Lyberg, L. (1977 b): Coding of Verbal Information. Memo, Statistics Sweden.

Lyberg, L. (1978 a): Automatisk kodning i HBU 78, Memo, Statistics Sweden (In Swedish).

Lyberg, L. (1978 b): Försök med automatisk kodning i hushållsbudgetundersökningen (HBU). Metodinformation, Statistics Sweden, no 78:2 (In Swedish).

Lyberg, L. (1979): Några experiment med automatisk kodning i AKU. Memo, Statistics Sweden (In Swedish).

Lyberg, L., Nordling, P. and Elmdahl, J. (1973): Kodningskvaliteten i lärarregistret. Memo, Statistics Sweden (In Swedish).

Lyberg, L. and Tillquist, B. (1974): Upprepad beroende kontrollkodning med consensus och Delphiteknik, några experiment. Memo, Statistics Sweden (In Swedish).

Lynch, B.T. and Arends, N.L. (1977): Selection of a Surname Coding Procedure for the SRS Record Linkage System. Memo, US Department of Agriculture.

McCalla, G.I. and Sampson, J.R. (1972): MUSE: A Model To Understand Simple English. Communications of the ACM, vol 15, no 1, pp 29-40.

Miller, J.G. (1975): Personal communication.

Minton, G. (1968): Some Observations Concerning a Model for Classification Control. Memo, US Bureau of the Census.

Minton, G. (1969): Inspection and Correction Error in Data Processing. Journal of the American Statistical Association, vol 64, pp 1256-1275.

Minton, G. (1970 a): Some Decision Rules for Administrative Applications of Quality Control. Journal of Quality Technology, vol 2, no 2, pp 86-98.

Minton, G. (1970 b): Comments on Quality Control and Research in Data Processing Programs. Paper presented at the American Society for Quality Control, March 12-13, Arlington, Virginia.

Minton, G. (1972): Verification Error in Single Sampling Inspection Plans for Processing Survey Data. Journal of the American Statistical Association, vol 67, no 337, pp 46-54.

Minton, G. and Krivitsky, Z.F. (1971): Two Additional Properties of a Model for Quality Control of Administrative Processes. Journal of Quality Technology, vol 3, no 1, pp 6-17.

Morgan, H.L. (1970): Spelling Correction in Systems Programs. Communications of the ACM, vol 13, no 2, pp 90-94.

Muth, Jr., F.E. and Tharp, A.L. (1977): Correcting Human Error in Alphanumeric Terminal Input. Information Processing & Management, vol 13, pp 329-337.

Newcombe, H.B., Kennedy, J.M., Axford, S.J. and James, A.P. (1959): Automatic Linkage of Vital Records, Science, 130, pp 954-959.

Norberg, A. (1976 a): Analys av frekvenser - en delstudie i projektet Automatisk kodning. Memo, Statistics Sweden (In Swedish).

Norberg, A. (1976 b): En urvalsfråga för lexikonkonstruktion. Unpublished working paper, Statistics Sweden (In Swedish).

Norberg, A. (1977): Några beräkningar kring en urvalsfråga för lexikonkonstruktion. Unpublished working paper, Statistics Sweden (In Swedish).

Official Statistics of Sweden (1969): Population and Housing Census in 1965, Part 9, Statistics Sweden (Summary in English).

Official Statistics of Sweden (1974): Population and Housing Census 1970, Part 12, Statistics Sweden (Summary in English).

Official Statistics of Sweden (1979): Population and Housing Census 1975, Part 11, Statistics Sweden (Summary in English).

Olofsson, A. (1976): Kvalitetskontroll av näringsgrenskodningen i AKU hösten -74. Memo, Statistics Sweden (In Swedish).

Olofsson, P.O. (1965): PM beträffande variabiliteten i närings-grens-och yrkesangivelser vid arbetskraftsundersökningar. Memo, Statistics Sweden (In Swedish).

O'Reagan, R.T. (n.d. a): Computer-Assigned Codes From Verbal Descriptions. Memo, US Bureau of the Census.

O'Reagan, R.T. (n.d. b): Address-Matching by Computer. Memo, US Bureau of the Census.

O'Reagan, R.T. (1967 a): Project to Study Potential for SIC Coding of Establishments on the Computer. Memo, US Bureau of the Census.

O'Reagan, R.T. (1967 b): Improvement in Computerized Address Coding. Memo, US Bureau of the Census.

O'Reagan, R.T. (1968 a): Test of Improvement in Computerized Address Coding. Memo, US Bureau of the Census.

O'Reagan, R.T. (1968 b): Application of Information Theory to Bureau Problems in Record Linking. Memo, US Bureau of the Census.

O'Reagan, R.T. (1972): Computer-Assigned Codes from Verbal Responses. Communications of the ACM, vol 15, no 6, pp 455-459.

Owens, B. (1975 a): Corbett Algorithm Implementation. Memo, US Bureau of the Census.

Owens, B. (1975 b): The Corbett Algorithm for Coding from Free Word Descriptions. Memo, US Bureau of the Census.

Palme, J. (1971): Personal communication.

Phillips, L.D. (1973): Bayesian Statistics for Social Scientists. Nelson.

Polya, G. (1976): Probabilities in Proofreading. The American Mathematical Monthly, vol 83, no 1, pp 42.

Rustemeyer, A. (1977): Toward Development of a Computer-Assisted Telephone Interviewing System. Memo, US Bureau of the Census.

Scott, W.A. (1955): Reliability of Content Analysis: The Case of Nominal Scale Coding. Public Opinion Quarterly, pp 321-325.

Silver, J. (1976): Evaluation of Geographic Coding for the 1972 Economic Censuses. Memo, US Bureau of the Census.

Sirken, M. (1973): Proposal for Modifications to the Verification Procedures for the Current Mortality Sample. Memo, US Department of Health, Education and Welfare.

Skotte, L. (1977): Personal communication.

Statistiska centralbyrån (1965): Nordisk yrkesnomenklatur (In Swedish).

Statistiska centralbyrån (1969 a): Svensk utbildningsnomenklatur (SUN). Statistiska centralbyråns broschyrserie, nr 18 (In Swedish).

Statistiska centralbyrån (1969 b): Standard för svensk näringsgrensindelning (SNI). Meddelanden i samordningsfrågor, nr 8 (In Swedish).

Statistiska centralbyrån (1973): Standarder i svensk statistikproduktion. Meddelanden i samordningsfrågor, nr 6 (In Swedish).

Sudman, S. (1967): Reducing the Cost of Surveys, Chicago, Aldine.

Sukhatme, P.V. and Seth, G.R. (1952): Non-Sampling Errors in Surveys. Journal of the Indian Society of Agricultural Statistics. 4, pp 5-41.

Sunter, A.B. (n.d.): Record Linkage. Unpublished memo.

Sunter, A.B. and Fellegi, I.P. (1967): An Optimal Theory of Record Linkage. Bulletin of the International Statistical Institute, vol 42, Book 2, pp 809-833.

Sveriges Offentliga Utredningar (1971): Utredning om lärarnas arbetsförhållanden (In Swedish).

Swensson, B. (1969): Alternativa surveymodeller för dikotoma variabler. Forskningsprojektet FEL I UNDERSÖKNINGAR, rapport nr 19, Statistiska Institutionen, Stockholms universitet (In Swedish).

Tappert, C.C. (1972): A Preliminary Investigation of Adaptive Control in the Interaction Between Segmentation and Segment Classification in Automatic Recognition of Continuous Speech. IEEE Transactions On Systems, Man, And Cybernetics, vol SMC-2, no 1, pp 66-72.

Tappert, C.C., Dixon, N.R. and Rabinowitz, A.S. (1973): Application of Sequential Decoding for Converting Phonetic to Graphic Representation in Automatic Recognition of Continuous Speech (ARCS). IEEE Transactions On Audio And Electroacoustics, vol. Au-21, no 3, pp 225-228.

Taubman, P. (1973): Occupational Coding. Annals of Economic and Social Measurement, pp 71-87.

Templeton, M.C. and Evans, M.C. (1970): Automated Classification of Medical Entities (ACME) for Selection of Causes of Death. Paper presented at the 98th Annal Meeting of the American Public Health Association, Houston, Texas.

Tepping, B.J. (1968): A Model for Optimum Linkage of Records. Journal of the American Statistical Association, vol 63, pp 1321-1332.

Tepping, B.J. (1969): The Application of a Linkage Model to the Chandrasekar-Deming Technique for Estimating Vital Events. Memo, US Bureau of the Census.

Thorelli, L.E. (1962): Automatic Correction of Errors in Text. BIT 2, pp 45-62.

US Bureau of the Census (1956): Sampling in the 1950 Census of Population and Housing. Bureau of the Census Working Paper No 4, Washington D.C.

US Bureau of the Census (1965): United States Censuses of Population and Housing, 1960: Quality Control of Preparatory Operations, Microfilming, and Coding. Washington D.C.

US Bureau of the Census (1972 a): Census Use Study. OS ADMATCH: An Address Matching System. Washington D.C.

US Bureau of the Census (1972 b): Evaluation and Research Program of the US Censuses of Population and Housing, 1960: Effects of Coders. Series ER 60, no 9, Washington D.C.

US Bureau of the Census (1974): Coding Performance in the 1970 Census. Evaluation and Research Program PHC(E)-8. US Government Printing Office, Washington, DC.

US Bureau of the Census (1977): 1976 Census of Camden, New Jersey, Memo.

US Department of Health, Education, and Welfare (1980): A Methodological Study of Quality Control Procedures for Mortality Medical Coding. Vital and Health Statistics, Series Z, no 81.

US Department of Labor (1978): Handbook of Occupational Keywords, 2nd edition.

Vossler, C. (1964): The Use of Context for Correcting Garbled English Text. Proceedings of the 19th National Conference, ACM.

Warner, S.L. (1965): Randomized Response: A Survey Technique for Eliminating Evasive Answer Bias. Journal of the American Statistical Association, pp 63-69.

Weatherall, J.A.C. (n.d.): Multiple Cause Coding - Exercise to Evaluate Coding Methods. Memo, Office of Population Censuses and Surveys, UK.

Weaver, L.A. (1975): Inspection Accuracy Sampling Plans. American Society for Quality Control, Technical Conference Transactions - San Diego.

Weiler, H. (1965): The Use of Incomplete Beta Functions for Prior Distributions in Binomial Sampling. Technometrics, pp 335-347.

Winch, R.F., Mueller, S.A. and Godiksen, L. (1969): The Reliability of Respondent-Coded Occupational Prestige. American Sociological Review, vol 34, no 2, pp 244-251.

World Health Organisation (1967): International Lists of Causes of Death.

Zadeh, L.A. (1965): Fuzzy sets. Information and Control, no 8, pp 338-353.

SCB-tryck, Örebro 1981